

Best Practices for Monitoring Digital Customer Experience (DCX)

From customers to code to containers, learn how New Relic can help you deliver a seamless digital customer experience for website and mobile users.

Table of Contents

OVERVIEW	03
SECTION 1. DIGITAL CUSTOMER EXPERIENCE MONITORING IN PRACTICE	04
Visibility across your full stack	04
Starting with your user experience	04
The website experience	05
The mobile experience	06
Check through your backend services	08
Harden your underlying infrastructure	09
SECTION 2: BUILDING PROCESSES ACROSS YOUR TEAM	10
Intelligent alerting for more actionable responses	10
Create deeper visibility into your technology, users, and business	10
Show the impact of technology on the larger business	11
Create a customer-centric view of your business	13
NEXT STEPS	14

Overview

More and more companies now interact with their customers largely via digital channels, making the digital customer experience (DCX) they provide a critical component of business success. The [New Relic Digital Intelligence Platform](#) is designed to help development and operations teams monitor the health of your technology stack, quickly troubleshoot any issues identified, and share key metrics of your digital business with a broader team of stakeholders.

In this guide, you'll learn best practices for optimizing the DCX of your websites and mobile apps as well as for improving their underlying services and infrastructure.

To begin, it's important to measure the service-level quality of your digital experiences across three dimensions:

1. **Availability:** Is it up and running?
2. **Functionality:** Is it working right?
3. **Speed:** Is it working fast enough?

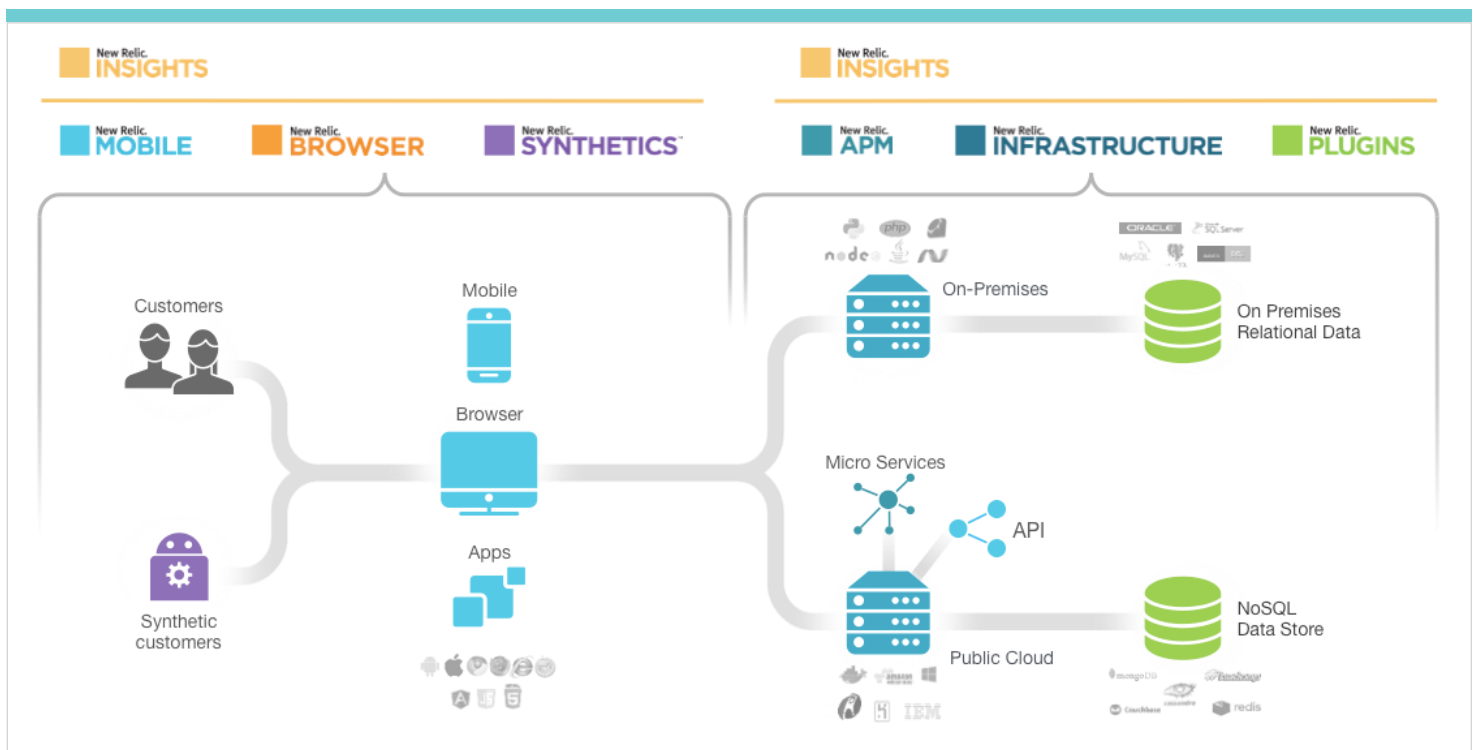
This creates a framework to improve the digital customer experience through the different layers of your technology stack.

Section 1. Digital customer experience monitoring in practice

Visibility across your full stack

The New Relic Digital Intelligence Platform is designed to provide visibility through all the different parts of the technology stack to help you quickly troubleshoot any issues that may be degrading the digital experience for your customers.

That's important, because your customers have no sense of whether it's the frontend, backend, infrastructure, or third party causing an issue—they simply suffer from a substandard experience. No matter the cause, the dissatisfaction generated by these disappointing experiences reflects poorly on your business.



Starting with your user experience

Given that the customer is the center of the experience, let's start with understanding the technology layer closest to the customer: the website frontend or native mobile app itself. We'll address websites and mobile apps separately because they have slightly different characteristics.

THE WEBSITE EXPERIENCE

Let's run through some of the key health metrics and features that New Relic offers to track the quality of your website's digital customer experience.

Availability: If your site is completely down, then functionality and speed have little meaning. That means the first goal is simply to verify your website is accessible:

- **Check browser throughput.** When you suspect something has gone wrong, the best test for overall application health is your **throughput**. Looking at your actual user traffic in **New Relic Browser** can reveal whether your customers are engaging with your site in real time. If this traffic has dropped off, it could mean users aren't able to reach your website at all, possibly due to a DNS or routing issue, or that they've navigated away from your site completely in frustration. An unexpected dip in throughput is often a sign that there's an issue to troubleshoot.
- **Set up availability alerts.** Since you don't want to be notified of outages by frustrated customers, you want to continually check that your website is up by using **New Relic Synthetics' availability monitors**. If a URL isn't reachable, a page isn't rendering properly, or an API is reporting incorrect payloads, New Relic can trigger a **monitor alert** and notify you before your customers are even aware of the problem. These monitors can regularly test your website from multiple global locations based on the distribution of your customer base. When coupled with the global throughput information in New Relic Browser's **filterable geographies**, you can get a more comprehensive view of your data and a more complete picture of availability than you could with Browser/Synthetics monitoring alone.
- **Track uptime service level agreements (SLAs).** Establish the baseline uptime for your website with the **SLA report** in **New Relic Synthetics**. These monitors keep tabs on your overall uptime on a daily, weekly, or monthly cadence, which you can communicate to stakeholders throughout your team, business, or customer base.

Functionality: Once you establish that the website is up, you need to make sure it's not broken for your customers:

- **Validate that key user actions work correctly.** Emulate real customers going through critical paths in your applications (i.e., login, checkout, directory, search, etc.) with New Relic Synthetics' **scripted monitors**. This will help you identify if something is explicitly broken for your users and to ensure your most important transactions are being exercised every minute from around the globe. The goal is to ensure that any customer-impacting issues, even in the underlying backend and infrastructure layers of your website, will also be detected.
- **Triage the JavaScript errors.** When something has gone wrong on the frontend, it's generally a black box since there are no available logs to parse. Review the exceptions your customers are experiencing in their browsers with New Relic Browser's **JavaScript errors**. To help prioritize errors for troubleshooting, you can see this information by name, website frequency, browser, or any other custom attributes you find important..

- **Set up functionality and error alerts.** As with the availability metrics, all of these metrics are integrated with New Relic's [alerting system](#) and can be set up to notify you when anything has gone wrong. Setting up alerts on your most important scripted monitors is particularly useful; notifications here mean a critical user path through your application has just been broken.

Speed: Now that you know you're up and running, it's time to make sure your website is not so slow that it's generating high user bounce rates:

- **Triage frontend load times.** Identify your slowest [page loads](#) in New Relic Browser to see the major bottlenecks in your applications. Leverage [percentiles](#) and [histograms](#) to better understand what's happening across your user base (a small set of pages can skew averages when viewed with less sophisticated tools). Couple this with custom targeting of your site's most important pages (checkout, login, etc.) to focus your efforts on the areas of greatest impact.
- **Optimize and verify frontend code.** Blocking JavaScript and un-optimized AJAX calls can lead to slow frontend execution. Use New Relic Browser to triage these issues, and then verify that your changes have actually improved the digital customer experience. Using [historical data](#) can help you easily identify bottlenecks, and then quickly and objectively quantify your engineering improvements.
- **Configure your frontend Apdex score.** New Relic uses an industry standard known as [Apdex](#), which categorizes your site's response times relative to a user-defined value. Set up an [frontend Apdex threshold](#) in New Relic Browser to match the responsiveness your customer base expects. Then work to improve this benchmark of user satisfaction.
- **Manage page bloat.** A major cause of slow page loads is the burgeoning size of web pages. [Waterfall graphs](#) in New Relic Synthetics can show you the size of the assets in your pages (i.e., images, media), while [session traces](#) in New Relic Browser show how long it takes for actual users to load these assets and execute JavaScript, with granularity down to individual users.
- **Check third-party JavaScript slowness.** Page bloat can also result from embedded third-party snippets and widgets: tracking, advertising, media, social, analytics, user chat, support, and more. These services can cause performance bottlenecks and may need to be disabled when they lead to site performance delays or breakages. Use [session traces](#) in New Relic Browser to identify the right third-party culprit. This can also inform discussions on the "value" of these widgets vs. the page load "cost" they incur.
- **Set up page load speed alerts.** [Dynamic baseline alerting](#) can help set the thresholds for healthy page-load response times. A gradual degradation of response times could be the precursor to more serious problems and can serve as an early warning signal, especially after a recent code deploy.

THE MOBILE EXPERIENCE

Here are some of the key health metrics and features that New Relic offers to track the quality of your mobile app's digital experience, again using the dimensions of service-level quality discussed above:

Availability:

- **Track app launches.** Perhaps the most important question for mobile developers is “Are people using my app?” **New Relic Mobile** answers this question by tracking how often an **app launches**. If this number collapses, especially after a new release or mobile OS version, it could mean users are experiencing errors, crashes, or slowness.
- **Review crash occurrences.** The frustration of opening an app to an immediate crash is often enough to make a user delete the app. **Crash analysis** offers detailed insights into why crashes are occurring in production, with the context needed to help fix them. You can better understand your highest-priority crashes through powerful analytics, filter to focus on high-crash screens, view the most common location in the code associated with crashes, or drill down to an individual user’s crashes.
- **Set up alerts.** If your number of app launches falls, or the number of crashes skyrockets, particularly after a new release, you want to be first to know. Set up **alerts on the health of your mobile app** to make sure you quickly learn of these customer issues, before your app store ratings start to sink with crash complaints.

Functionality:

- **Fix HTTP errors.** Rarely are mobile apps self-contained. Instead, they’re typically dependent on backend APIs, but limited visibility makes it difficult to debug API errors. Shared context into the backend with **HTTP request** information can equip mobile developers to be more responsive to API errors and partner more closely with backend teams. **Cross-application traces** help teams understand the end-to-end path of an HTTP transaction from the mobile app to the corresponding application. Combined with New Relic Synthetics’ **API monitoring** of the backend, you can proactively identify issues before customers experience them.

Speed:

- **Improve HTTP response times.** In mobile development, you can only be as fast as your backend APIs. When silos separate mobile and backend developers, performance SLAs are hard to establish and audit. Using New Relic Mobile, you can see the **API transactions** broken down by location, device, and even connection type. A common language and data showing where milliseconds are being spent helps make performance a responsibility for everyone on the team.
- **Track user interaction times.** A poorly written piece of mobile code can cause screen stutters, or even lock up a UI thread completely in a hard freeze. New Relic Mobile builds an analysis of the speed of each user interaction in your app, with detailed breakdowns of the most **common and slowest interactions**. That helps you spend your time where you can make the largest impact, leveraging real interaction data with granularity down to individual users.

Check through your backend services

Websites and native mobile apps are often powered by a set of supporting APIs and microservices. If these backend services degrade, so can the digital customer experience. Fortunately, there are techniques that can make it easier to monitor and debug these backend components to help ensure a consistently high-quality experience for your customers:

- **Ensure the health of your backend APIs and microservice topology.** Identify your key microservice bottlenecks using [New Relic APM service maps](#) and tune their performance. Then set up [API monitors](#) in New Relic Synthetics on these services to validate that the services are available, providing the correct API payload responses, and doing so within the response time defined by your SLA. Microservices that are not publicly accessible can be monitored using [private minions](#) in New Relic Synthetics.
- **Configure your backend Apdex score.** Set up an [Apdex](#) threshold in New Relic APM to match the responsiveness from your customer base, and align this to your [Apdex score in New Relic Browser](#). You will likely find the greatest opportunity to improve the speed of your page load in the frontend, which is monitored by New Relic Browser. However, this optimization work will often involve jumping into the backend to tune slow endpoints and services.
- **Check performance of third-party services.** Third party-services for data, email, messaging, content, and other functions can create backend dependencies. Any issues with your third-party vendors can cause degradations or crashes; monitor these services and endpoints with [New Relic Plugins](#) and [New Relic Synthetics API monitors](#)—and decouple these external services if necessary.
- **Triage backend errors.** Review the errors most common in the most important parts of your application errors in [New Relic APM error analytics](#) so you can triage and address the ones with the greatest impact on your quality of service.
- **Identify key transactions in your services.** After identifying your key services, you can start identifying the key application transactions within these services—the most critical path through your most critical services. Flag them as [key transactions](#) in New Relic APM for heightened monitoring visibility to help ensure early awareness of any issues.
- **Resolve slow application transactions and database calls.** Troubleshoot slow-running application transactions or blocked database calls with the greatest impact on your customers for tuning and optimization. Triage these bottlenecks in [New Relic APM transactions](#) and [databases](#), respectively, to check overall responsiveness and digital experience.
- **Set up backend application alerts.** If your application server goes down, your whole digital experience goes down, so be sure to set up alerts on the most critical areas of your application so you can be notified of the warning signs:

- Set up uniquely prioritized alerts on the **key transactions** so that if they break, you know to expedite your response.
- Configure **deployment markers** in New Relic APM to provide greater context to your alerting policy. Alerts triggered by broken key transactions, spikes in error rates, or lengthening response times immediately after a deployment may point to a bad release that needs to be rolled back.

Harden your underlying infrastructure

The foundation of your technology stack is your on-premise, hybrid, and/or cloud infrastructure. Issues at this tier can bring down everything above it, including your digital customer experience. The measures below can help make sure you have full visibility into how your infrastructure is working:

- **Make sure hosts are responsive.** Create **'Host Not Reporting' alerts** in New Relic Infrastructure to find out if you have unresponsive systems.
- **Track system resource health and consumption.** If your underlying infrastructure goes down, your whole digital experience goes down, so it's critical to set up **New Relic Infrastructure alerts** to ensure key resources like compute, network, and storage are not exhausted or over-provisioned, or that vital processes have not gone down.
- **Track configuration changes.** Infrastructure changes are a leading cause of outages. Use **correlated health metrics** in New Relic Infrastructure to make sure your containers and infrastructure don't go down due to a bad configuration update, even across different data centers, hybrid environments, or cloud providers.
- **Manage capacity and scaling.** Leverage **load testing** to see if you have sufficient infrastructure resources to handle traffic spikes without service degradation as well as for capacity planning for future growth.
- **Understand performance of third-party services.** Third-party services for data stores, load balancing, caching, messaging, queuing, and others can create backend dependencies. Vendor issues can cause degradations or crashes; for example, you can monitor AWS services and endpoints with **New Relic Infrastructure Integrations**.
- **Track performance of Docker containers.** Container services create an additional technology layer where system issues may arise. Review **Docker health metrics** in New Relic to ensure system stability.
- **Integrate monitoring into your IT automation workflow.** Make sure your **configuration management tools** include monitoring as part of your infrastructure automation to ensure resources are automatically instrumented for monitoring. New Relic Infrastructure, for example, offers pre-built integrations for **Ansible**, **Chef**, and **Puppet**.
- **Review security updates and packages.** Zero-day vulnerabilities and outdated packages can expose security vulnerabilities, and can require extensive inventory audits to patch. Quickly search across your infrastructure in seconds using the **inventory search** function in New Relic Infrastructure to address these security issues.

Section 2: Building processes across your team

The complexity of modern technology stacks creates a tremendous amount of surface area to manage and monitor. To make digital experience monitoring effective at scale, you need to highlight the most urgent items with actionable context, route that information to the correct teams, and integrate it into your everyday processes. Let's walk through putting it all together:

Intelligent alerting for more actionable responses

Alerts are the critical step for creating actionable notifications for key events throughout your technology stack. Building out a comprehensive and intelligent alerting policy helps teams deliver a better digital experience—and sleep more soundly at night:

- **Create a better signal-to-noise ratio.** Noisy alerting policies create alert fatigue and desensitization to notifications by “crying wolf.” That’s a recipe for truly meaningful alerts to be missed until it is too late. [Dynamic baseline alerting](#) can help establish the healthy band for key metrics, even considering seasonality, cyclicity, or noisy data patterns. Use [severity-level thresholds](#) to prioritize high-importance notifications and prevent desensitization to alerts. Group alerts together into [defined incidents](#). And be sure to provide [runbook instructions](#) so responders understand the potential risks of the alert and how to respond appropriately.
- **Create more targeted alerts.** [NRQL alerting](#) lets you build code-defined alerting policies so you can create more targeted alert criteria on important metrics. For example, you can create alerts on specific high-priority response codes, custom error codes, specific application exceptions, system metadata, cloud infrastructure tags, and more.
- **Route issues to the accountable team.** Update [notification channels](#) and [alert contacts](#) for faster resolution to make sure the alert gets sent to the right team first, depending on what part of the stack is affected.
- **Integrate alerts into ChatOps and existing workflows.** Pre-built alerts integrated into [collaboration platforms such as Slack and HipChat](#) notify engineers where they hang out, with additional available integrations and webhooks into your existing collaboration workflows, enterprise service buses, custom systems, and more. New Relic also offers [iOS and Android apps](#) to view dashboards and receive push notifications.

Create deeper visibility into your technology, users, and business

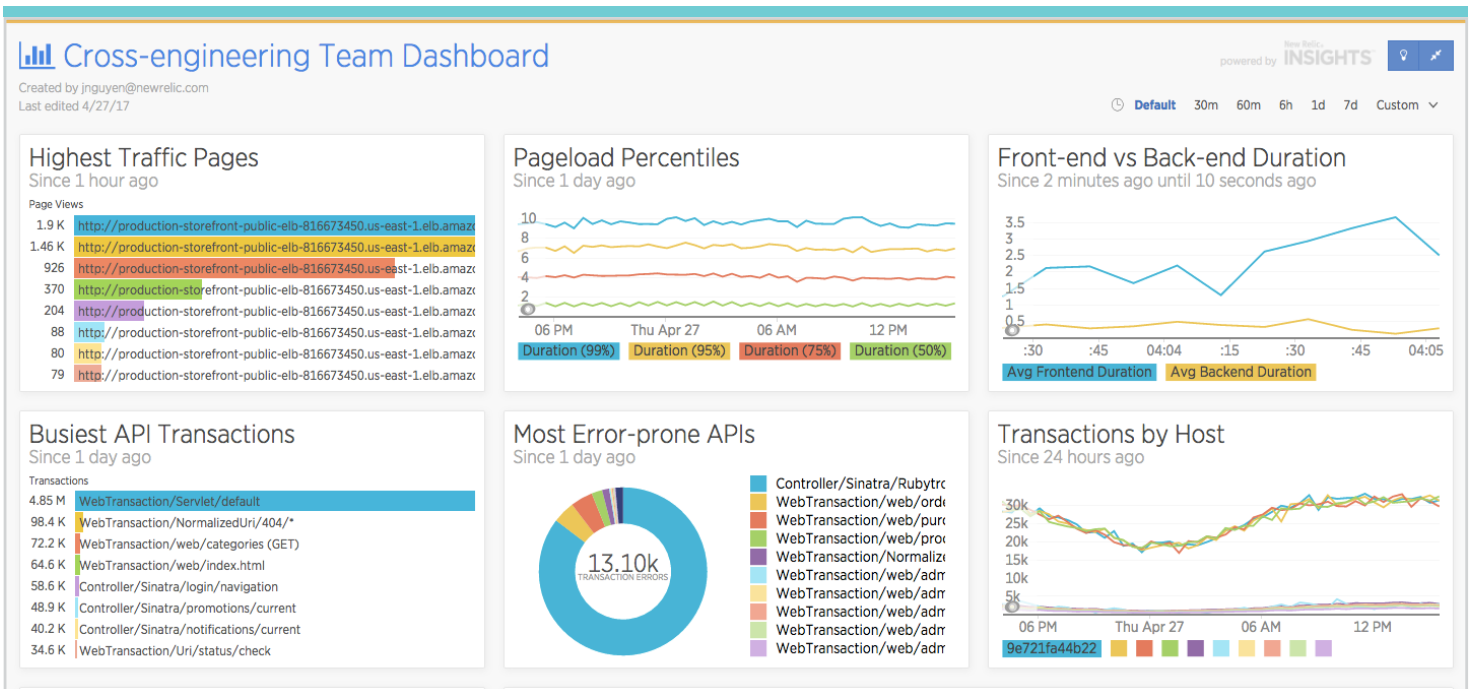
The specifics of delivering digital customer experiences are unique to different technology stacks, users, and businesses. Often, there are custom metrics specific to your business that should be included in your performance monitoring.

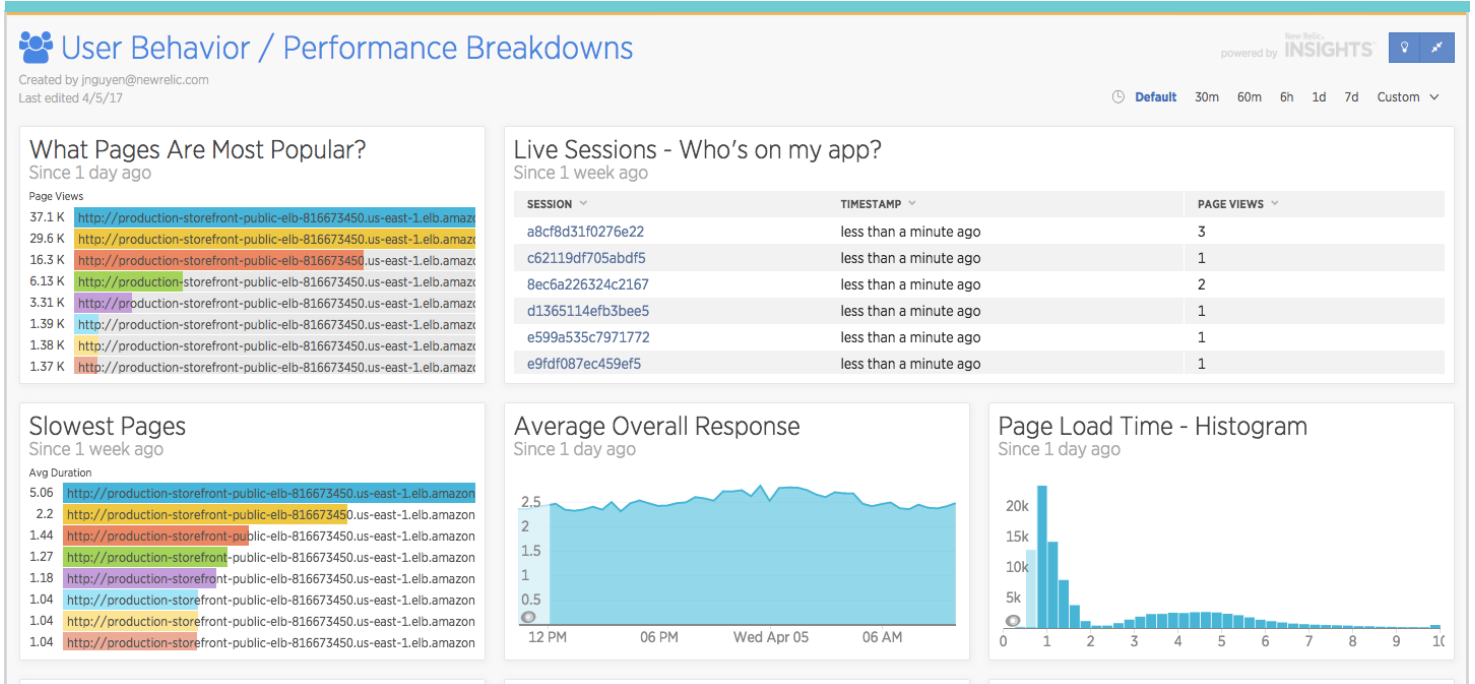
Monitor additional values about your digital experience. Use **custom events**, **metrics**, and **attributes** in **New Relic Insights** to monitor additional dimensions. There are several data types to include:

	People	Things	Money	Technology
Data type:	account_id, user_email	product_id, content_type	product_value, user_LTV	deployment_id, release_version, feature_name
Benefit:	Pivot performance information around your users	Track product and content consumption and engagement	Understand revenue at risk, cost of outages, financial impact	Know the exact code and context to recreate bugs more quickly

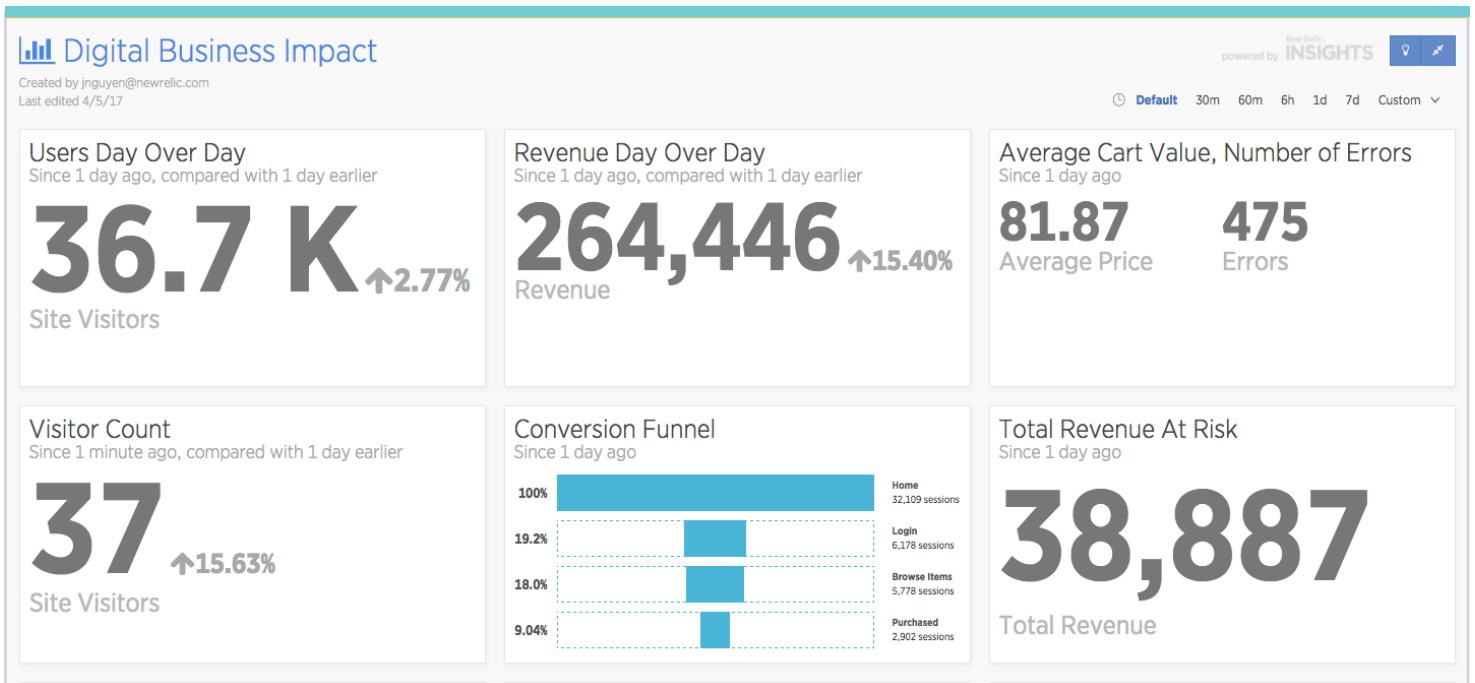
Show the impact of technology on the larger business

The quality of the digital experiences delivered by engineering teams matters. The way your website and apps run affects the way your users engage with them, which in turn drives the overall business. Align different engineering teams together to improve the performance of your digital experience, and demonstrate the fruits of their efforts to the broader business, using shared real-time dashboards and reporting. Build these visualizations in New Relic Insights to bring together all the performance data across your stack—as well across your custom dimensions—to show how it all connects together.





Customer experience dashboards show how performance impacts user engagement



Business dashboards show how the overall digital experience drives business outcomes

Create a customer-centric view of your business

The structure of your organization or the layers of your technology stack can heavily influence the ways you build and run your digital customer experiences. However, all of that is irrelevant from the customer perspective—they should never need to care about what’s happening behind the scenes. Making sure everything works together seamlessly is the heart of delivering a great digital customer experience.

- **Frame performance around customers, not technology.** With a `userId` custom attribute, New Relic Insights can pivot performance data around your digital customers, not just your technology stack. This customer-centric view focuses the conversation on delivering the best digital experience, instead of on how your engineering teams are organized.
- **Stack rank your VIP customers and high-value accounts with live performance dashboards so you know who they are and the quality of their digital experience.** Your most important customers give you the most money and use your products the most. Yet, in some cases, they may have the worst overall experience due to personalized settings, custom code, or more sophisticated usage. Leveraging a customer-centric view of how your digital experiences are performing can bring into focus how different functions can better serve your most important customers:

Development	Prioritize VIP tickets and reproduce those specific errors more easily with detailed customer data
Operations	Provide SLA uptime reports to help uphold any contractual sales obligations for VIP accounts
Product	Tailor the product roadmap and understand feature usage by power users
Support	Prioritize VIP tickets and proactively reach out to key customers concerning any service issues
Customer Success	Build deeper relationships with high-value customers leveraging better understanding of their usage and experience
Marketing	Align account-based marketing initiatives to customer engagement and satisfaction
Sales	Understand customer engagement with digital channels for better account management
Executives	Strategically plan around key customer relationships, account growth, and consumption shifts

Next steps

We encourage you to apply the best practices outlined above. Start with improving the frontend performance your customers are experiencing. From there, move to the backend and down to the infrastructure. Set up alerts through the process, and track custom events to build out custom dashboards that speak to your organization's particular needs. Finally, be sure to share these dashboards across your extended teams to drive conversations on how the digital customer experiences you deliver impact the bottom line.

If you need more resources to get the most out of the New Relic Digital Intelligence Platform, we're here to help. [New Relic University](#) and our [Technical Documentation](#) offer deep learning and technical information. For even more guidance, our fee-based [Expert Services](#) team can help you extend New Relic to meet your specific requirements—so you can glean actionable intelligence into your particular customers' experiences and needs.

Finally, remember that creating a great DCX requires more than just technology. As website and mobile apps become strategic differentiators for companies, they're transforming the roles and responsibilities of the teams that build and maintain them.

You can see that transformation in the rise of [DevOps](#), a multidisciplinary approach well suited to the importance and complexity of building a world-class DCX. Data-driven DevOps, along with the agile development practices that typically go along with it, can help multiple parts of the organization work together to more quickly [create and deploy higher-quality, more secure, and more reliable software](#) that delights customers and meets business objectives. And that's what delivering a great digital customer experience is all about.

New Relic is a leading digital intelligence company, delivering full-stack visibility and analytics to over 40% of the Fortune 100. The New Relic Digital Intelligence Platform provides actionable insights to drive digital business. Companies of all sizes trust New Relic to monitor application and infrastructure performance so they can quickly resolve issues, and improve digital customer experiences. Learn more at newrelic.com.