



Traffic, Trade, and Timing

How to mitigate risk when peak web traffic is more than it can handle

What Will You Learn:

- The imperative.
- Reasons for peaks.
- What happens when a website gets more traffic than it can handle?
- Mitigating the risk – and maximizing the opportunity.



The Imperative

Most businesses consider more web traffic to be a good thing. It's a measure of success. Of visibility. It means they're being found on search engines, that their marketing campaigns are working.

It is, then, easy to become so focused on the goal that you become blind to the risk. Too much traffic can also be a problem, and website outages can be highly costly in terms of both revenue and reputation.

If a website slows down noticeably or fails altogether, what started as a good news story can all too easily turn into a crisis. In this document, we look at ways to mitigate the risks – and make the most of the opportunities – associated with extreme traffic peaks.

Too much traffic can also be a problem, and website outages can be highly costly.



Reasons for Peaks

Websites see peaks in traffic for a number of reasons, some of which are easier to plan for than others.

Popular events

When you promote major events on your website, the demand brings a new level of traffic from people desperate to sign up. Your challenge is to deliver a consistent customer experience free from outages. The worst thing that can happen is having concert fans hoping to snap up tickets from your website and being left to stare at a message saying the site is unavailable.

Marketing campaign

In theory, a surge in traffic resulting from a big marketing campaign should be easy to plan and test for. Predicted visitor numbers should be factored into the campaign plan. However, there are risks. Communication is a potential issue.

- Does marketing know that traffic could be a problem?
- Have they spoken to their colleagues in IT?
- Will everything be ready in time for testing and, if necessary, remediation?

There is also an additional risk in the sense that businesses rely on such campaigns for revenue. A drop in revenue and wasted investment on advertising at the very time when revenue was predicted to rise will have a significant impact on the business.

Media coverage

The effects of a news story or social media sensation are by their nature harder to predict. Getting caught up in the hype (positive or negative) of a big news story may be unlikely, but the timing and extent of any traffic-related issues are unknowable in advance. It is therefore important for every organization to assess the extent of the risk and consider how far it is a) worth mitigating and b) possible to mitigate. For example, is it in a high-profile sector? Are there foreseeable newsworthy events that could have an impact on the organization? What would be the cost of any outage? And how does that compare with the cost of maintaining redundant capacity or implementing a more scalable solution?

Product launches and collection releases

Collection releases and product launches draw mass interest and buzz from buyers that can lead to website slowdowns and unsatisfied customers voicing their frustrations on social media channels. During these events the website is a key channel for the online releases and can be pushed to its limits to meet the demands of the increase in popularity.

A large volume of traffic could result in spikes in CPU usage on a web server or cause it to run out of memory.

Website and business launches

If you're starting from scratch, it is of course difficult to predict how many visitors you can expect. Past data is going to be limited or non-existent. The risk is also high. A big failed launch may be hard to recover from – even if the failure itself may attract some helpful (if not entirely positive) publicity.

Organic growth

It is possible for organizations to be caught out simply through organic growth in traffic. While it's easy for a business to see that it has outgrown its physical premises, it may be harder to see that the infrastructure for its website is no longer up to the task. This is where monitoring – either synthetic or real user – is invaluable. This should reveal any gradual slowdowns and/or minor outages well before the tipping point at which the website becomes unusable or fails altogether.

One thing to remember

One thing to remember when forecasting traffic based on a previous event (such as a Black Friday promotion) is that your analytics might not be telling you the full story. If your site slowed down (or had an outage), the number of people who tried to access your site will be higher than the reported number of actual visitors.

WHAT WILL YOU LEARN

What Happens When a Website Gets More Traffic Than It Can Handle?

Sometimes, it's the hardware itself that runs into problems. A large volume of traffic could result in spikes in CPU usage on a web server or cause it to run out of memory.

Websites that use shared hosting solutions may be particularly vulnerable because their servers have to deal with traffic from multiple websites – someone else's peak may become your own.

There are other possible bottlenecks. For example, on a database-driven website, high levels of traffic may cause the maximum number of concurrent connections to be exceeded.

It is therefore important for ops teams to monitor every part of a system that could come under stress during testing or during a real peak event.



Locking mechanism on the database system

One typical bottleneck of the e-commerce application resides in its transaction database. This locking system synchronizes with the inventory system, places items the customer seeks to purchase on hold and reserves them for a specific amount of time during checkout.

If the customer decides not to complete their purchase in the designated time frame, the lock is removed, and the item can be reserved by the next customer in line. This entire process can cause major slowdowns, as these types of database transactions must be centralized.

Since online retailers are often dealing with limited quantities of physical merchandise for seasonal sales or product launches, you must preserve the database integrity for order and stock volumes.

If a shopper just bought the very last item, the website should immediately reflect the item is out of stock for the next shopper. The effort in preserving the data integrity calls for the high unification of system requirements, which can have negative impacts on website performance.

The worst case scenario of database system failure is the security threat of shoppers' personal and credit card data being compromised. If the database isn't functioning properly and these calls are not delivered at the speed of the requests, orders can be mixed up, resulting in the same item of inventory being sold multiple times.

Controlling website traffic and throughput based on the speed that this locking mechanism can operate is the only way to safeguard against the massive influx of customers attempting to reserve inventory of limited or deep-discount items.

If the database can handle the increased rate of inventory locking, congratulations! However, the next technically complicated step, checkout, represents another potential bottleneck during your high attraction event, due to the increased conversion rate, and your rate of sale (the speed at which customers are trying to checkout) of the peak traffic.

If the customer decides not to complete their purchase in the designated time frame, the lock is removed, and the item can be reserved by the next customer in line.

Payment systems not processing the increase in transactions

Another common bottleneck point for online transactions occurs at the payment system. Most credit card services are provided by third-party vendors via payment gateways, which perform a variety of technical checks and verifications to process secure financial transactions.

These verifications come at a cost, and the latency of completing the transaction can average from 2–3 seconds on a typical day, but can also slow to 5–10 seconds at high traffic times, all depending on the numbers of payment authorizations needed.

This sheer amount of time combined with the variability of time required contribute to instability when the payment system is pushed to its transactional limits.

Beyond the time the payment verification requires, payment gateway providers have a finite number of transactions they can handle. So, on Black Friday or iPhone release day, when ten times the number of transactions are expected to be handled by the same payment gateway service and infrastructure as a typical day, something may well go wrong.

These problems are not limited to small businesses with their own hosted websites or lower tier subscriptions with e-commerce platforms, but also impact the ability for enterprise-level companies processing a considerable number of transactions.

WHAT WILL YOU LEARN

Mitigating the Risk – and Maximizing the Opportunity

There are a number of ways to reduce the risks of high volumes of traffic on a website. However, these shouldn't just be thought of as ways to avert disaster – steps such as optimizing a site for performance will also help you capitalize on a surge in visitor numbers, delivering the best possible experience when it matters most.

Eggplant's performance experts can test your site, so you know how it copes under load.



Testing

The only way to understand how your website behaves under load is to try it in controlled conditions and monitor the results.

This is where performance testing comes in.

Performance testing comes in a variety of different flavors, and the right kind for you depends on the pattern of traffic you're planning for. Here are a few of the most common:

Load testing involves emulating the traffic you're expecting. Typically, this means ramping up traffic levels gradually, just as a marketing campaign would, for example, gain traction over time. Most organizations build in a high degree of redundancy to allow for the fact that it's impossible to forecast with complete accuracy. It's also common to look at how traffic is expected to increase as a result of general business growth to make it easier to plan ahead. Although load testing is a subset of performance testing, the two phrases are commonly used interchangeably.

Stress testing will keep piling on the pressure to identify the points at which the application slows down and eventually fails. It can therefore help organizations understand whether and indeed when they will need to invest in added compute power.

Spike testing tests what would happen in the event of a sudden flood of traffic, over and above expected levels. This helps organizations understand how the system will respond to major, unexpected events or denial of service (DoS) attacks.

Soak testing (or endurance testing) is designed to eradicate issues that only come to light when a system is experiencing elevated levels of traffic over a prolonged period. Most organizations need a combination of tests. For example, even when traffic can be predicted with a reasonable degree of confidence, nothing is certain. Stress testing is often carried out at the same time as a load test. The results of a stress test may be useful to have if plans change or if forecasts suddenly have to be revised upwards.

Whatever the type of testing, it's important to note that the aim is not to mimic the traffic itself, but the effects of that traffic. It is not, for example, necessary to test every conceivable user journey through a site, but to select journeys that test the system in a way that's representative of real traffic. The aim is to test key business functionality and architectural features.

Timing

When testing in advance of a planned event, timing is critical. On the one hand, it's important to test as early as possible to allow as much time as possible to identify and fix any performance issues – as well as a retest to determine whether any such remedial work was successful.

However, testing too early comes with its own set of problems. It might be too soon to get an accurate prediction of traffic. Content or systems might also be subject to changes that would render the results of the test all but useless.

Staging, pre-production or live

Another important decision to make is what kind of environment to test on. In an ideal world, it should be as 'like-live' as possible. Testing on a live environment comes with obvious risks and drawbacks, and while out-of-hours testing can reduce those risks, this isn't always possible (for example, if the site has an international audience).

A test on anything other than live may be less representative of the real thing, but it is far easier to control and far less risky.

Monitoring during testing

Knowing when an application slows down or fails is important, but it's also important to understand the root causes. It's therefore useful to make the most of built-in server monitoring tools to help identify the bottlenecks. It can also be useful to have an application performance management (APM) solution in place during testing.

At the same time, some kind of external monitoring – either synthetic or real browser – may be helpful, both in terms of tracking down the source of any problems and in terms of showing the actual impact on the end user.

Removing content

Some organizations deliberately serve slimmed down versions of web pages at peak times. This can mean less traffic hitting the web server than would otherwise be the case. By the same token, it can be a good idea to build ad landing pages from a different template from the rest of the site. This helps to ensure you deliver only the content you need for that page, without making visitors download lots of styles and scripts they may never need. As a result, your pages will be both faster and more resilient.

Eggplant's monitoring insights and real customer insights solutions include performance impact prediction feature. Using historical data to tell you how a change in load times will affect conversion, bounce rate, pages per session and revenue.

Using real customer insights to predict the business impact of a slowdown during peak

However, delivering lighter pages is good practice even outside of peak periods. A small page will load faster than a big one thus minimizing content means minimizing the resources on which the website depends – whether it's bandwidth or infrastructure. So when that unexpected peak does hit, a lighter website should be that much better able to cope than one that's bloated with content.

It's therefore important to audit websites regularly and remove redundant content. Unfortunately, this kind of routine maintenance can slip down the list of priorities, meaning that organizations gradually store up problems for themselves as they constantly ratchet up the volume of content. For example, it's not uncommon to find duplicate script libraries, even on large, otherwise well-maintained sites.

Third-party management

Websites are increasingly reliant on thirdparty content. Retail sites in particular depend on a whole range of third-party services, such as analytics, multivariate testing, re-targeting, payment gateways and social media integration.

Any one of these could slow down or fail under load, just as your own website can – with two added complications.

One is that your visibility of and control over third-party infrastructure is limited. You can't test it and you can't see how it behaves under test. That's not to say it doesn't get tested – your suppliers are every bit as interested in keeping their services up and running as you are.

The other complication is that in most cases your website will just be one of many to rely on these services. You have no control over other peaks that could affect them. For example, even if Black Friday is irrelevant to your business, you could effectively be dragged into traffic-related issues over this period because of others who rely on the same third-party services as you do.

Web performance experts at Eggplant can help you optimize your website as you prepare for peaks in traffic with performance health checks and pre load test performance audits.

Auditing

The first step in managing third-party content is to audit it – to find out what is on your site and gauge its importance. Some services, such as payment solutions and quoting engines, are integral to the running of a website – it just wouldn't work without them. Others may be regarded as optional extras.

The best approach to each service depends on a) its value to the business and b) the impact on the website if it were to fail. A third-party service on this retail site slowed down significantly on Black Friday 2016.

Single points of failure

A single point of failure (SPOF) is anything that would effectively take the site down or make it unusable if it failed to load. Many third-party services are now implemented in such a way as to minimise the possible impact on the host site (for example, through asynchronously loaded or deferred JavaScript).

However, others, such as multivariate testing or feature detection services, may have to be loaded up front, presenting a risk of a SPOF in the event of a traffic-related slowdown. For this reason, some retailers opt to suspend such services during planned peaks.

More generally, since you can't know how each service will behave during peaks, it's important to shop around. Select services that would have the least serious impact on your site if they were to fail, moving away from suppliers with riskier implementations.

Managing traffic

One increasingly popular way to reduce the risk of traffic from a marketing campaign is to spread the load over a longer time-frame. This might involve staggering email campaigns or running a promotion over an extended period to avoid a flood of traffic in one hit.

Queuing solutions

Virtual waiting rooms are very useful both for performance and for risk mitigation. By adding a virtual waiting room system to your website, end users exceeding your website capacity limits are offloaded to the virtual waiting room.

When space opens up, the virtual waiting room redirects the end users who waited in line back to your website in the correct, sequential order and at a pace your site can handle.

If you are planning an online sale event that will attract significantly higher traffic or a significantly higher conversion rate, a virtual waiting room system will be key to avoiding major technical issues and customer uproar. This is particularly useful for organizations that don't want to invest in scaling up their infrastructure for a one-off event.

Caching

A website's content can be cached in several ways, but the basic idea is always the same: store content in a more convenient location or in a more convenient way. This results in less traffic to and/or less work on the web server.

Caching on the web server

Server-side caching is important for database-driven sites, where the server has to query a database to build pages. If it has to do this for every visit to every page, this can cause considerable strain. A better approach is for the web server just to query the database for the first visit, then cache (store) the plain HTML page. When the next visitor accesses the same page, the web server can just deliver the page it has in its cache, without having to build it again.

Queue-it enables you to gain control of website overload during extreme traffic peaks by offloading end users into an online queue.

Content delivery networks (CDNs)

CDNs are very useful both for performance and for risk mitigation. They work by storing copies of the resources needed for a website in various locations across the world. End users can then retrieve the resources they need from the location that's closest to them, rather than getting them from the origin server.

This is particularly useful for sites with an international footprint, where high latency can be a real problem. It also means that less traffic finds its way back to the origin server, allowing the site to cope with higher levels of traffic.

Other types of proxy cache

A CDN is an example of a proxy cache – in other words, it behaves like an intermediary between the end user and the site owner. There are other kinds of proxy cache, though. For example, ISPs may intercept and cache content rather than send requests back to the origin every time.

This saves them bandwidth and can also help that content get to the end user faster. The difference is that CDNs are under the site owner's control, while other types of intermediary are not. However, site owners can still take advantage of proxy caches by setting future Expires headers for static assets (things that don't change very often) and avoiding marking them as 'private' (non-cacheable by intermediate caches).

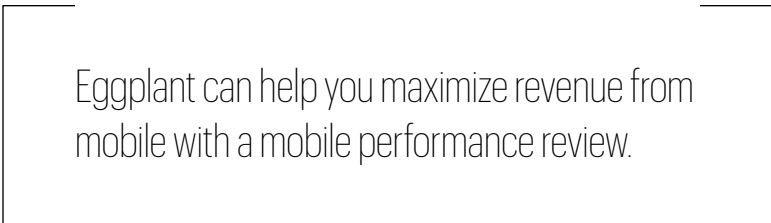
Browser cache

Again, this is something that is only under limited control of the site owner, as end users may decide to clear their browser cache at any time, and mobile devices will tend to have less storage available.

However, it's still important for site owners to adopt a coherent caching strategy. For example, this may involve setting long cache lifetimes and avoiding generic file names (e.g. `main.css` or `heroimage1.jpg`) for static assets.

Other forms of client-side storage

There are other ways for web applications to store data on end users' devices. For example, the rise of progressive web apps means that `localStorage` and `IndexedDB` are likely to become more popular. In a progressive web app (essentially a website that delivers a more app-like experience on mobile), a service worker can intercept a network request to deliver content from the device rather than retrieve it over the Internet. This allows the app to work offline, even though it won't have access to the most up-to-date content.



Eggplant can help you maximize revenue from mobile with a mobile performance review.

The mobile factor

Shopping on mobile phones has become incredibly popular, and most retailers are well aware of the fact that they need their website to work on mobile.

However, recent trends mean there are hidden dangers that could trip site owners up.

For example, m dot sites are increasingly rare. In many ways, this is a good thing. It should mean that sites are a bit faster on mobile because mobile users are saved the cost of a redirect from the main domain.

Mobile-specific sites have largely been replaced with responsive sites. On a responsive site, the content simply reflows to fit the viewport. This helps create a seamless experience between desktop and mobile, and also saves retailers from having to develop and maintain two separate sites in parallel.

However, responsive sites can be slow on mobile. This is because they tend to deliver much the same content as they do on desktop. In the worst cases, big hero images designed for desktop screens are simply resized or cropped to fit into smaller boxes, forcing end users to download large files they don't really need.

Fortunately, it is now possible to write HTML that will select the right image for the viewport, but not everyone has taken advantage of this still relatively new standard. Some sites, for example, still use JavaScript to select different images for mobile. However, this comes with its own set of problems, not least of which is that JavaScript tends to be slower to process on mobile, particularly lower-spec devices. Other sites do use the responsive images specification correctly, but even this doesn't guarantee a fast experience on mobile, as some designers and developers will use it to deliver large, high-resolution images to devices with high device pixel ratios.

The mobile factor summary

This will mean superbly crisp imagery, but it could come at the cost of speed, which is critical during peak.

So as you approach a peak trading period, it is a good idea to:

- Audit responsive imagery on your site – are you just scaling images down or are you using smaller files for smaller viewports?
- Consider reviewing or even suspending 'DPR switching' – the practice of delivering high resolution images to devices that support them.
- Test on lower-spec mobile phones – not all smartphones are equal, and the last thing you want to do during peak is turn a significant proportion of your customers away.

Load balancing

Load balancing involves routing traffic to different servers on a distributed system in order to minimize the load on any one server. This can be done in a number of ways. At its simplest, load balancing just shares traffic across servers evenly. More intelligent load balancers can avoid sending traffic to a server that is struggling or slow to respond for whatever reason. It's even possible for a load balancing service to add extra servers to a system in response to increasing load. This is a form of cloud bursting – increasing capacity by taking advantage of additional resources in the cloud.

Cloud bursting is one way for small organizations with limited infrastructure to deal with peaks, but it comes with its own challenges. For example, it can take time for those extra resources to respond to the increase in traffic. For this reason, cloud bursting is usually better suited to planned peaks or peaks with a gradual ramp up than it is to sudden spikes in traffic.

Contingency planning

What if it all goes wrong? What if, despite your best efforts, your website fails under the weight of traffic for one reason or another? Mitigating risk is one thing, but eliminating it entirely is impossible. It's therefore critical to have contingency plans just in case things go wrong.

For example, will someone always be on hand to investigate and fix any issues? And how will the organization deal with any negative press coverage? Will helpline or in-store staff be able to deal with frustrated customers?

Peaks in traffic aren't just a technical challenge, and they require a joined up approach that involves the whole business. For this reason, it's important to get senior management buy-in and for leaders within the organization to ensure that the relevant departments are able to respond in a coordinated way.



Keysight enables innovators to push the boundaries of engineering by quickly solving design, emulation, and test challenges to create the best product experiences. Start your innovation journey at www.keysight.com.

This information is subject to change without notice. © Keysight Technologies, 2021 – 2023, Published in USA, January 19, 2023, 7121-1164.EN