How a modern frontend can transform your ecommerce business

5 ways JAMstack and a modern CDN edge cloud platform will boost your ecommerce frontend.
Introduction

FinTech, ecommerce and other online businesses operate in very competitive and fast evolving sectors. Customers expect smooth, fast and intuitive online experiences and are always looking for the latest or best sites to invest their time and money in.

This is also a critical time for businesses. Between the continued growth of ecommerce and the COVID crisis, ecommerce grew by 46% in 2020, according to Internet Retailing. This growth has primarily benefited the best-in-class digital organizations, forcing others to accelerate their digital transformation in order to stay competitive and ensure they are positioned to best take advantage of future opportunities.

Your frontend is critical when it comes to delivering the level of experience that customers expect. If you are too slow making changes to your web apps, or workflows, it becomes harder to convert or retain customers. It doesn’t take much to start falling behind the competition, which in turn requires spending more money on external providers and undertaking complex system upgrades to try and make up lost ground.

Monolithic systems ruled the digital industry for years, but have failed to keep up with the pace of innovation. Combining the power of microservices, APIs, cloud, and headless technologies, is the answer.

Today, modern front-ends are powered by JAMstack architectures and advanced edge cloud platforms. They boost online performance, improve security, bring agility to development teams and scalability to your overall ecosystem.

Getting ahead on the modern web

Before you start, it is important to make sure this kind of digital transformation is right for your business. There are two key factors to consider:

• Is there a clear business case?
  How fast is your site? How does the overall experience on your competitors’ sites compare to your own? How does your target customer shop, and how does that affect the performance requirements on your existing site in terms of site speed, search, personalisation, and content, etc.? You need to understand these factors in order to accurately assess whether you actually require the flexibility of a headless or microservice based approach, or if your goals could be supported fully with a more traditional approach. Going headless can come with its own complexities, so don’t look to implement it just for the sake of it. There needs to be a clear strategy behind it, a thorough understanding of why you are implementing it and how it can help achieve what’s on your roadmap.

• Do you have the right IT skills and depth of resources?
  To effectively operate a headless platform, you need the right operational infrastructure. Ideally, you want to be able to harness cloud technology and manage the integration of multiple SaaS products. Other key capabilities include being able to automate deployments and testing, provide a service mesh, security management tools, and access development and test environments for a multiple application footprint where potentially you previously only had one (if you are moving from an all-in-one platform, like SAP or Oracle Commerce).
The Modern Frontend

A modern front-end centers on the customer, not the technology, as explained by the MACH Alliance. JAMstack architectures and edge cloud platforms are the two core technologies powering this evolution:

- **JAMstack** is a modern web development that decouples the frontend presentation layer from the backend business logic.

- **Modern CDN edge clouds** enable further reduction of lag time & infrastructure costs, increases scalability, and improves security. They also enable intelligent microservices routing, content personalization, as well as provide full API visibility and advanced security protection.

The combination of these two technologies on your front-end architecture boosts performance, shortens time-to-market, and frees business growth. JAMstack's API-focused architecture enables merchants using any ecommerce platform to add and remove external tools at any time, without impacting site performance or the system's integrity. Edge cloud platforms bring a load of features for optimizing performance to deliver better end-user experiences.

Because every component is pluggable, scalable, replaceable, and can be continuously improved through agile development, "it enables faster feature development and innovations, more joined up personalized and localized experiences, and more performant and reliable customer experiences", as highlighted by the Mach Alliance.
Here are 5 ways that the modern stack will positively impact your business operations.

**END USER** | End users get to enjoy a faster, more personalized online experience.

**EDGE CLOUD PLATFORMS**
An edge cloud platform improves user experience by reducing lag time & infrastructure costs, increasing scalability, and improving security. It enables intelligent microservices routing, content personalization, provides full API visibility and advanced security protection.

**JAMSTACK FRONTEND**
A modern web development architecture based on client-side JavaScript, reusable APIs, and prebuilt Markup to boost frontend frameworks. It provides shortened time-to-market and free business growth, as its API-focused architecture enables merchants that use any eCommerce platform to add and remove any external tools at any time, without jeopardizing the site's performance or system's integrity.

**WEBSITES & APIs**
Application Programming Interface is a software intermediary that allows two applications to talk to each other. Each time you use an app, send an instant message, or check the weather on your phone, you’re using an API.

**BACKEND**
The Backend includes the software that creates the actual website pages, the shopping cart and the product search engine, and an electronic data interchange (EDI) system than enables online customers to receive real-time inventory and order status.

**DATA LAYER**

**ENTERPRISE SYSTEMS**
1. Deliver your content faster & save on operations costs

According to the HTTPArchive metrics report, users wait an average of 2.5 seconds for primary content to load on desktop, and 5.2 seconds on mobile. If we are talking about engagement with a page, time to interaction goes up to 9.1 seconds on mobile. Headless architectures based on JAMstack, coupled with a CDN edge cloud, speed up time to first byte and page load times significantly.

A JAMstack setup serves sites as static assets. Since all the markup and assets are pre-built, they can be cached and served via CDN, avoiding the extra round trip to the origin server for content. This will considerably boost the delivery of your assets, providing a better user experience and easier scalability for your apps and sites.

Working alongside your JAMstack app, a modern CDN edge cloud further boosts performance and speed. It caches content closer to users and collapses similar requests into one for fewer trips to origin. Best in class CDN edge clouds can also cache API responses because content can be programatically purged across the network in a matter of milliseconds. Visitors get a better and faster experience from your headless architecture, and you benefit from less traffic to your API servers with greater predictability and uniformity of requests to origin.

Finally, because hosting static files is cheap or even free, the JAMstack architecture automatically cuts some of your operating costs.

2. Offer more personalized experiences

The decoupling between the presentation layer and content management in the JAMstack approach means content delivery can be treated as yet another service to plug into. Since personalization is inherently dynamic, it is relatively simple to augment how content is served to users with standalone personalization services that hook into applications via serverless functions and web hooks. These mechanisms allow content marketers to give their users immediate feedback and vary content based on individual customer characteristics. Serving content statically on a CDN has the additional benefits of being able to access features like A/B testing at the edge so that granular metrics can be measured without an added trip to the server.

Edge cloud platforms can also make use of edge dictionaries that act as a distributed database at the edge composed of key-value pairs. This enables you to classify visitors and direct them to a specific experience, which is useful if you want repeat visitors to have a different experience than first-time visitors to further increase engagement.

A serverless computer environment also enables you to build applications and execute code at the edge — without having to manage the underlying infrastructure. Your teams can deploy prebuilt edge modules or write custom code to take advantage of content targeting, A/B testing, content stitching and more, and thus further personalize responses to each individual user by serving as much content as possible from the edge within microseconds.

3. Make your ecosystem more secure and get better monitoring capabilities

Traditional server-side and CMS applications expose lots of APIs and services to allow administrators to manage data and content. All of these are at risk of attack and require continuous effort to patch and maintain. A JAMstack architecture, on the other hand, relies on static files which are read-only and not susceptible to the same types of attacks. There is no code to run, so no vulnerabilities to exploit.

Adding a CDN edge cloud will further secure your applications. For instance, you can move authentication to the edge for increased performance and security. Managing TLS & API calls from the edge will allow you
to provision, upload, and manage tens of thousands of
certs on behalf of your end-users while keeping this
process out of your origin, which makes security easier
and more automated. A CDN edge cloud will also shield
your origin from DDoS attacks by collapsing all similar
requests into one, so your origin never gets overloaded
if an attack does occur.

Finally, CDN edge cloud also provides real time logs
from the edge, enabling you to monitor in real-time
request response times, origin failures, or suspicious
traffic and feed data into the logging endpoint of your
choice for further analysis. This provides visibility into
how visitors are engaging with your content, helping
you to identify trends and resolve any API delivery
problems faster. You can monitor the impact of new
code deployments or API versioning and, in the event
of an issue, roll back to a previous stable configuration
in seconds.

4. Free your development team & scale
your operations

Deploying a headless frontend will probably be
welcomed by your development teams. Free from the
rigidity of a monolithic infrastructure, they can choose
which framework or programming language to work with
and build really rich web experiences. Relying on APIs
to manage workflows and content also usually means
a shorter learning curve along with the flexibility to
change out any provider at any time.

Scale is another key element. Can your app or site
ingest a tenfold, a hundredfold or a thousandfold of its
normal traffic, if suddenly everyone wants to go to it?
Because JAMstack relies on containers to host data
and content, it is easy to scale and add the necessary
resources within seconds.

CDN's load balancers can also support the needs
of Layer 7 load balancer allows you to define content-
aware routing decisions while ensuring instant
convergence and failover. Unlike DNS-based solutions,
you get immediate and granular control. As an extra
bonus, CDNs also provide improved performance and
cost savings over ADCs, especially during flash traffic.

5. Streamline data representation and
simplify the backend-frontend relation

The continued growth of digital commerce means that
ecommerce applications are becoming increasingly
complex. You may be utilizing different microservices for
customers, orders, products, shopping carts and other
service functions, all of which expose APIs to be used by
the frontend. But offering a more personalized service
also often means having different client interfaces
with different needs that might depend on the same
underlying resources, such as web and mobile clients or
experiences for new and returning customers.

The Backend for Frontend (BFF) pattern has emerged as
a response to this challenge. Here each client application
has its own server-side component, so that you can shift
a large part of your front-end logic to an intermediate
layer. Rather than a single monolithic API backend, each
frontend has its own API layer. This means that your
backend no longer has to spend energy on response
markup generation, making it simpler to manage the flow
of information between APIs and customers.

When combined with an edge cloud, BFF can help
speed up development processes as well as site and
app performance. With each microservice having a
separate BFF, teams are free to add or remove data
from particular apps without the need to change core
APIs. Web and mobile teams, for instance, can work
separately to create better experiences and add new
functionality without having to worry about impacting
the other. It can also help improve security by allowing
sensitive data to be hidden and unnecessary data to
be omitted entirely when a response is returned to
the frontend.