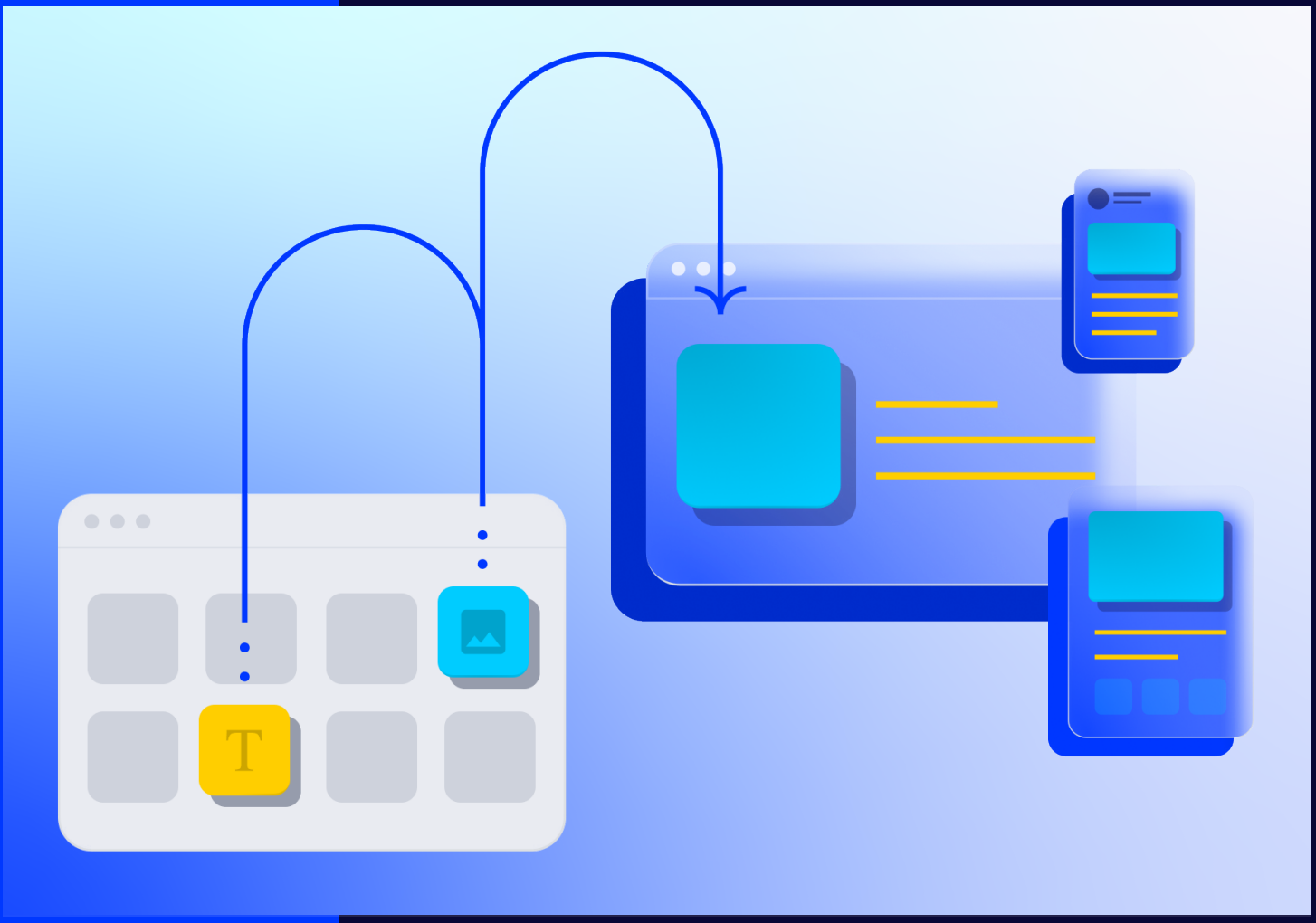


GUIDE

The essential guide to headless CMS

Your guide to understanding the definition, evolution, benefits, and drawbacks of a headless CMS

Optimizely

Contents

01
Headless defined

Is someone on your team looking for a CMS? Wondering whether you need a headless solution? You're in the right place.

This guide will help you navigate the world of headless by breaking down key industry terminology, fast-forwarding through the technology's evolution, and showcasing its benefits and drawbacks. And because your business is unique, we're here to help you understand whether a headless system is right for you.

As more and more companies faced the need to deliver content across multiple channels, the emergence of headless CMSs came as no surprise. Content management strategists have increasingly adopted headless to reach customers on new touchpoints, or to stitch together a complicated landscape of siloed applications. But first things first: before we go through the evolution, benefits, and drawbacks of a headless CMS, let's start by defining the key terminology.

When you research headless CMS, you'll most likely run into various terms connected to hosting models and delivery architectures. **It's essential to note that these are not the same thing.**

Hosting model: This is the environment your software or website runs in. On-Premise, Software-as-a-Service (SaaS), and Platform-as-a-Service (PaaS) are all common examples.

Types of hosting models

ON PREM

PAAS

SAAS

On-Premise (on-prem) hosting
Software is installed and managed on an in-house server, with the vendor simply selling and supporting it.

On-prem hosting is like owning your own house. You can do whatever you want with it and any issues that arise, well, you take care of them.

Platform-as-a-Service (PaaS)
Software is delivered over the internet, providing developers with the freedom to focus on building their solution without worrying about operating systems, software updates, storage, or infrastructure.

PaaS is like owning a condo. It's in a building owned by someone else, but you can do whatever you want to your own unit.

Software-as-a-Service (SaaS)
Instead of installing software, users access SaaS solutions over the internet through a centrally hosted application managed by a third-party vendor. Most SaaS applications can be run directly through a web browser, which means no installations are necessary.

SaaS is like renting an apartment. Your landlord owns both the building and your unit.

Delivery architecture is completely different. It's basically the framework for managing and delivering your content. Examples of delivery architecture include headless, coupled, and decoupled.

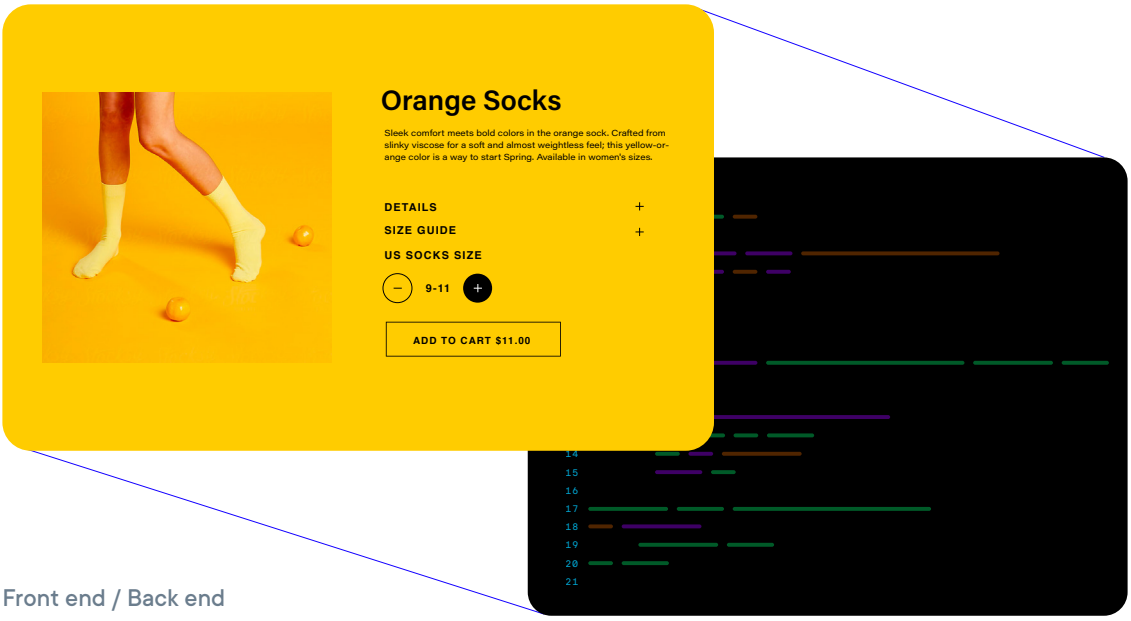
Just to be clear, management involves everything you do to the content before you publish it, while delivery covers the processes and tools that convert that content into something consumable and distributes it to your audiences.

Before we delve any deeper, let's first define a few terms that will be useful:

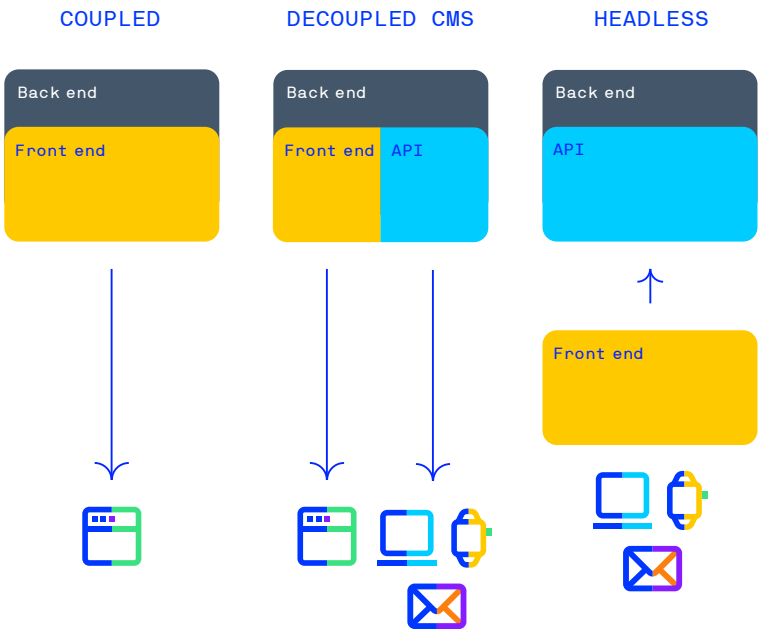
Front end: The front end, or presentation layer, is everything a user sees and interacts with. Traditionally that relates to a website, but today it can mean pretty much any device with an internet connection. Think images, text, layouts, etc. HTML, CSS, and JavaScript are three common front-end development languages you've probably come across at some point.

Back end: The back end is the developer's side of the website where servers, applications, and database information are all managed. So, rules, integrations, and the way pages connect are controlled here. Common back-end development languages include PHP, SQL, and Java.

Head: Simple, it's just another name for the front end of the website.



Front end / Back end



The front end and back end of a website need to work in harmony to create the end result – which is the delivery architecture in action. They can work together in a coupled, decoupled, or headless manner.

Coupled CMS: A traditional CMS architecture in which the front end and back end are bound together. In other words, content is created, managed, stored, and delivered in the same system. This is the most common architecture for web content delivery.

Decoupled CMS: No prizes for guessing that the front end and back end work separately. Content is prepared in the back end, then delivered to and presented in the front end through APIs. A decoupled CMS enables content managers, editors, and designers to work on the front end, while developers take care of the back end.

Headless CMS: A CMS architecture free from front-end components like templating and other frameworks. A headless CMS is a content-only data source without a presentation layer. So yes, the head, or front end, is removed from the body, or back end. In practice, headless is simply just another form of decoupled architecture, but instead of the CMS pushing content to the front end, the front end pulls content from the CMS.

> Not all headless CMSs are created equally. Some are pure headless, while others are a mix of a headless and traditional architecture.

There's more to headless than that...

Pure Headless: A pure headless CMS delivers pure data into the system. It doesn't transform it into any rendered, consumable asset.

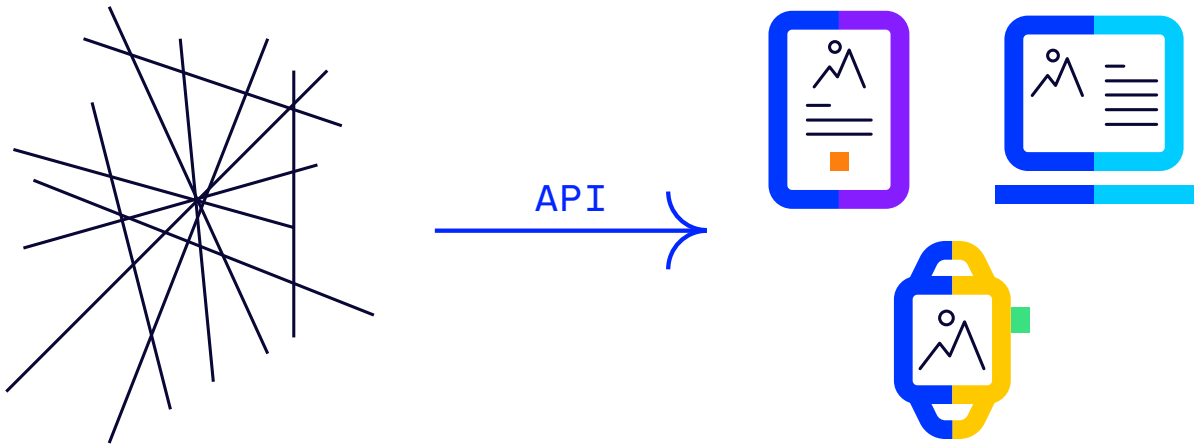
Hybrid Headless: A system that can operate in both a coupled (traditional) and headless CMS.

We mentioned APIs above, so let's make sure we understand that too.

APIs: An **Application Programming Interface** defines how systems interact with each other and allows them to communicate. APIs are essential to headless CMS. Without them, you wouldn't be able to pull content into another presentation layer, nor receive commands from users.

Okay, still with us? Here's a short summary so far:

Hosting models and delivery architectures are not the same thing, so you can mix and match. For example, you can deploy PaaS with a traditional delivery, PaaS with a headless delivery, SaaS with a headless delivery, and so on. In the context of digital experiences, headless lets you use your CMS to power various customer-facing experiences, including native mobile apps, smart devices, and existing web applications not built directly within the platform itself.



02

The journey to headless CMS

Content management first emerged with the creation of static websites in the 1990s and their simple HTML text files. Web 1.0 anyone? But with the introduction of new and more advanced technologies over time, content management had to keep up. Fast forward to...

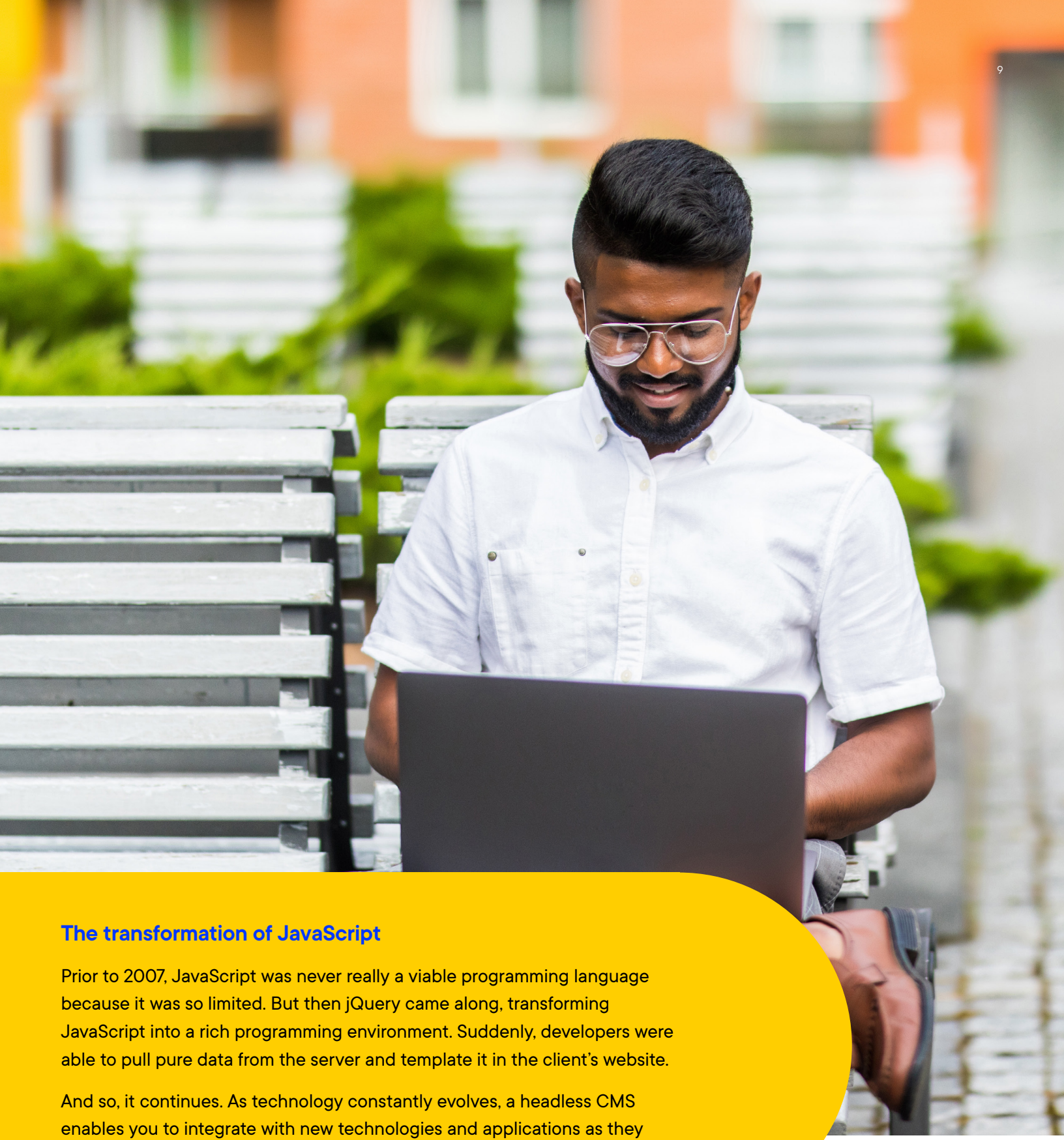
The mobile revolution

For years, your typical CMS claimed to deliver multichannel publishing, meaning you could push your content into any channel. But since most companies were only creating websites, they only had one channel. Mobile then came along to ring in the changes.

With the advent of mobile technology, other channels suddenly became far more viable. CMS technology needed to make good on its promises and enable users to push content wherever they wanted. As it turns out, this was a challenge since content management systems were really only created for delivering web content to desktops and laptops.

The rise of omnichannel publishing

Omnichannel publishing was the driving factor behind the rise of headless CMSs. A coupled CMS was no match for the demands of social networks, apps, and smart tech like watches and voice-activated devices. The flexibility required to support multiple channels accelerated the move to decoupled and headless solutions.



The transformation of JavaScript

Prior to 2007, JavaScript was never really a viable programming language because it was so limited. But then jQuery came along, transforming JavaScript into a rich programming environment. Suddenly, developers were able to pull pure data from the server and template it in the client's website.

And so, it continues. As technology constantly evolves, a headless CMS enables you to integrate with new technologies and applications as they emerge. And that's why it has become such a common delivery architecture.

03

The benefits of a headless CMS

Developers often push for headless due to its flexibility and the advantages offered by APIs, CDNs, and SDKs. But what benefits does a headless CMS deliver to your overall business?

BENEFIT 1

Integrate with other technology

Does your company run countless legacy web applications and mobile apps to power various phases in the customer journey? Rather than spend hundreds of thousands of dollars swapping out these legacy systems with a traditional CMS, a headless CMS will integrate all your current and future tech. Just think: no more monolithic applications resulting in silos that are expensive to manage and fail to engage customers.
A headless strategy will unify your content and enhance customer journeys.

BENEFIT 2

Reach customers on emerging touchpoints

As smart home devices become the norm, you need a CMS that can keep up with emerging technologies. Headless can integrate the underlying framework used by the smartest of smart devices and deliver digital experiences your end users expect.

BENEFIT 3

Respond to changing customer expectations

The cross-functional front-end code of a headless CMS creates continuity in the user experience. In theory, that helps you hit your targets more quickly, while also making you more responsive to changing customer expectations.

04

The drawbacks of a headless CMS

Remember that not every headless CMS is the same. So be warned: while pure headless can definitely solve many issues, many systems can also introduce new challenges.

DRAWBACK 1

Pure headless does not always equal simplicity

It may be initially appealing and straightforward to translate your content into raw data consumed by any platform. However, headless does not always resolve complexity; it often just moves it around. Saying a headless CMS is simpler is like a car manufacturer saying their cars are more reliable because they removed the engines. Headless CMSs are often described as simple because they focus on a limited feature set. Companies often find they need more functionality than is built-in to the CMS, so they have to customize.

DRAWBACK 2

Pure headless is not always faster or cheaper

Contemplating a headless solution vs. a hybrid solution? It's essential to consider the time to value and the total cost of ownership. Adopting a pure headless CMS means your engineering team takes on more responsibility to build out different models and templates. While this provides greater levels of control and customization, it also takes time to see value. You have to identify exact requirements, design models and modules, and then actually build them. Plus, you also need to invest more in ongoing engineering support to maintain your system, which drives up ownership costs.

DRAWBACK 3

Pure headless is not always right for marketers

Since headless removes the presentation layer shipped with the application, most platforms won't support essential native tools, like in-context editing or content preview before publishing. Yet marketers and merchandisers often use the same presentation layer as their audiences to edit, preview, and publish changes to the live website. Take away the ability to do that and they are left flying blind with one hand tied behind their backs.

DRAWBACK 4

Pure headless may not future-proof your CX

If you go for a headless CMS, chances are it won't include key technologies, like personalization, experimentation, and optimization, which all help you differentiate your customer experiences both now and in the future.
Some teams may prefer to build or buy these technologies as additional systems, but will need to integrate them with the CMS. They will also need clarity of design, so marketers can understand the technology that powers each element of the experience.

05

The big question: Is headless right for you?

Ultimately, you need to ask yourself one single question: will a headless solution help your organization close the gaps in your customer experiences faster, cheaper, and more efficiently than the alternatives?

Keep a cool head

As we've just seen, headless architecture comes with both pros and cons. And so, just like any major business decision, you have to perform your due diligence and match the requirements of your business with the deliverables of a content management system.

To put it another way, it's pretty much impossible to sum everything up in one quick, hard-hitting guide. So instead of helping you choose in five simple steps, we'll do you one better. Our resident content management expert, Deane Barker, has written an entire book dedicated to the subject.

Things You Should Know:
25 Lessons I've Learned About Selecting Content Technology and Services
by Deane Barker

It's a great resource and relays many of the lessons he's learned over 25 years of participating in the technology selection process as a buyer, technology vendor, and service provider. Sure, it's a longer read but it'll walk you through everything you need to know about selecting the right CMS from an expert in the industry.

06

Headless use case

The National Rugby League brings 19 websites and 7 digital properties into one seamless, efficient content hub for millions of visitors every week.

About The National Rugby League

The National Rugby League (NRL) is Australia's top-level domestic men's rugby league competition. It contains clubs from the original Sydney club Rugby League competition, which has been running continuously since 1908.

The Challenge

With 19 websites and 7 non-web digital properties, NRL's decentralized technology infrastructure and unintegrated tools were holding them back in several ways:

- Disruptive workflows, especially for the content team
- Inability to develop and optimize content across multiple digital properties
- Lack of analytics for data-driven decisions around content production
- Poor traffic and engagement due to blanket content creation

The Solution

NRL transitioned its sites to the unified, robust, and cost-effective Optimizely CMS. An integrated network and intuitive user experience made it easier for NRL to create and publish content swiftly, especially when it came to breaking news. Optimizely Search & Navigation made it easier for NRL to locate assets, and the team also leveraged data from the platform to better understand their visitors.

34%

YoY increase in user base and a significant increase in fan engagement on the NRL app

The Results

With over 3 million average weekly users, NRL saw a 34% YoY increase in user base and a significant increase in engagement from fans via the NRL app. Data has become an essential part of NRL's online engagement efforts, with every piece of content now tailored according to audience segment.

It has been a game-changer for us centralizing our architecture around the Optimizely CMS platform. It helped the organization to focus on work that really matters rather than managing complex technology architectures.

Now we can better analyze visitor data, generate insights, and improve our future and current digital initiatives to fit user demands. All the content we produce is much more targeted and valuable, driving higher engagement. We couldn't have done this on our old systems, moving everything to a single platform has been critical."

Quanah McBride
General Manager – Digital Technology & Operations, National Rugby League

07

Optimizely. The best of headless. Without the headaches.

A modern, extensive set of capabilities and REST-assured APIs transforms Optimizely into a true headless CMS. In doing so, we deliver all the flexibility you need to create, deploy, and manage a unified experience throughout all channels and devices.

Optimizely Content Graph streamlines the way you access your content across channels. Powered by GraphQL, it lets you search, query, and deliver content anywhere. Plus, you can enter any search term into the on-demand, fully-indexed content library to pull the exact content you need, in the format you need... without the headaches. It's a world away from simple parametric search or filtering. And because Content Graph runs on the edge, you can access your content super-fast, with no fuss.

The **Content Definitions API** simplifies the way you manage content and property definitions. There's no need for any coding.
And the **Content Management API** makes it a cinch to push new content into the Optimizely CMS.

When you're good to go, just switch on **multi-channel content mode** and check it out with a clean, content-first view. Not 100% happy? No problem. It's easy to create and edit folders, content blocks, and media. And if you're running more than one website, **multisite management** brings everything together in a single web app for a code base.

Our model is also unique in the way it supports both headless and coupled websites. The capability to easily decouple front-end and back-end means you can take advantage of advanced front-end technology, yet still retain the flexibility to deliver the most up-to-date content, wherever it needs to go.

It's important to note that a headless CMS is not necessarily a replacement for a traditional CMS. Some of our peers with outdated development practices, such as annual releases and ancient APIs, perpetuate the myth that a traditional CMS is no longer fit for the modern world. Optimizely shatters that myth with a continual release strategy and modern APIs that have kept pace with emerging technologies.

As a matter of fact, headless is an additional approach the Optimizely platform offers to cater to all business needs. Optimizely didn't build a new CMS from scratch since its core web CMS product provides a flexible base for an alternate delivery method. In other words, Optimizely is headless when you need it to be. And a fully-featured coupled CMS when you don't.

At Optimizely, we're on a mission to help people unlock their digital potential. With our leading digital experience platform (DXP), we equip teams with the tools and insights they need to create and optimize in new and novel ways. Now, companies can operate with data-driven confidence to create hyper-personalized experiences. Building sophisticated solutions has never been simpler. Optimizely's 900+ partners and 1100+ employees in offices around the globe are proud to help more than 9,000 brands, including Toyota, Santander, eBay, KLM and Mazda, enrich their customer lifetime value, increase revenue and grow their brands. Learn more at **Optimizely.com**