The
ULTIMATE
Guide to CMS

Traditional vs Headless CMS

Vol 01

TABLE OF CONTENTS

Traditional vs. Headless CMS

INTRO

The choice of your Content Management System (CMS) is of a much higher importance today than it has ever been. A traditional CMS worked well in the past because it only had to manage data and present it to a single platform – a browser on the web. Back then, efficiency trumped flexibility. However, the world has changed drastically and serving content only to a website is no longer enough.

A CMS needs to contend with a multi-device world-spanning web, mobile, tablets, smartwatches, digital assistants, AR/VR, and the Internet of Things (IoT). Moreover, a surging demand for dynamic and personalized digital experiences has marketers and developers scrambling for a new content management solution that can accommodate multiple-device delivery, but more importantly, one that can keep up with rapid change and please demanding audiences. Today, efficiency demands flexibility. This is where a "headless CMS" comes in.

WHAT IS A HEADLESS CMS?

A CMS typically consists of a frontend and backend. The "head" in CMS refers to the frontend, such as the templating system that includes the site's structure and style controlling the design of content, such as formatting, navigation menus, and forms that users see on a website. The backend contains parts that aren't visible to the user, like the system's programs, software, databases, and access controls.

Think of it like a restaurant. The frontend is what customers see and experience: the menu, the food and its presentation. Whereas, the backend consists of the kitchen, the appliances and personnel that work behind the scenes to create the experience for dining guests.

In a traditional content management system, the frontend and backend are tightly connected and intertwined into a single unit. If the "head" deals with the presentation of content, then a "headless CMS" does without the burden of a presentation layer and removes its outdated tether to a backend.

> **In other words, headless removes the connection between viewing and accessing the information and operates them like separate systems,**

linked by an application programming interface (API) that is designed to let them communicate and deliver content and data between one another, without being inherently dependent on (or coupled to) each other.

A headless CMS is also referred to as being an API-first CMS. A headless architecture opens up the content management system to a world of new potential, especially for omnichannel delivery that is to say delivering content formatted for display on multiple devices; such as a website, mobile phone, tablet, smartwatch, Internet of Things (IoT) devices.

WHAT IS DECOUPLED CMS?

While "headless" and "decoupled" are both departures from the legacy approach to content management, they are not identical. The term decoupled means that the back and front-end are not connected to each other through a database. Instead, they communicate with the front and back-end through APIs (but there is still a front-end).

When most people talk about a decoupled system, they are referring to a system that includes front-end delivery tools that are a part of the CMS, so templates and other formatting tools are available as a part of the system. The front-end is still handled through API controls, so it technically is still a decoupled CMS and you can manage, scale, tune and secure both systems independently from each other.

It can get confusing since vendors frequently use the term 'decoupled' to describe 'headless' and 'hybrid' CMSs. Often, it is not immediately obvious as to whether the system is a pure headless CMS or a CMS that includes formatting support.

> **A pure headless CMS is one that does not include the presentation layer of front-end code and formatting tools**

Suffice it to say that a pure headless CMS is one that does not include the presentation layer of front-end code and formatting tools, whereas a decoupled CMS typically includes presentation code and formatting tools.

DIFFERENCES BETWEEN TRADITIONAL AND HEADLESS

Headless CMSs have picked up steam in the last year; as a matter of fact, many Fortune 500 companies have already adopted the technology. However, there are a good number of organizations that question, "Do I need a headless CMS?" To answer this question, the following sections examine the architectural differences between the different types of CMSs, explore the pros and cons of each, and provide examples of when using a headless CMS can yield superior results. It also reviews a real-world headless CMS solution, Contentstack.

First, we'll tackle the differences between the traditional and headless CMS architectures.

Traditional CMS Architecture

In a traditional CMS (Adobe, Drupal, WordPress, SDL Tridion, Sitecore, and so on), having a front and back-end that are intricately bound together means that everyone, regardless how different their work is, is sharing the same workspace. Marketers and content editors create and publish content, while IT and developers program and manage the database and system applications in the backend of the same system that website visitors are viewing.
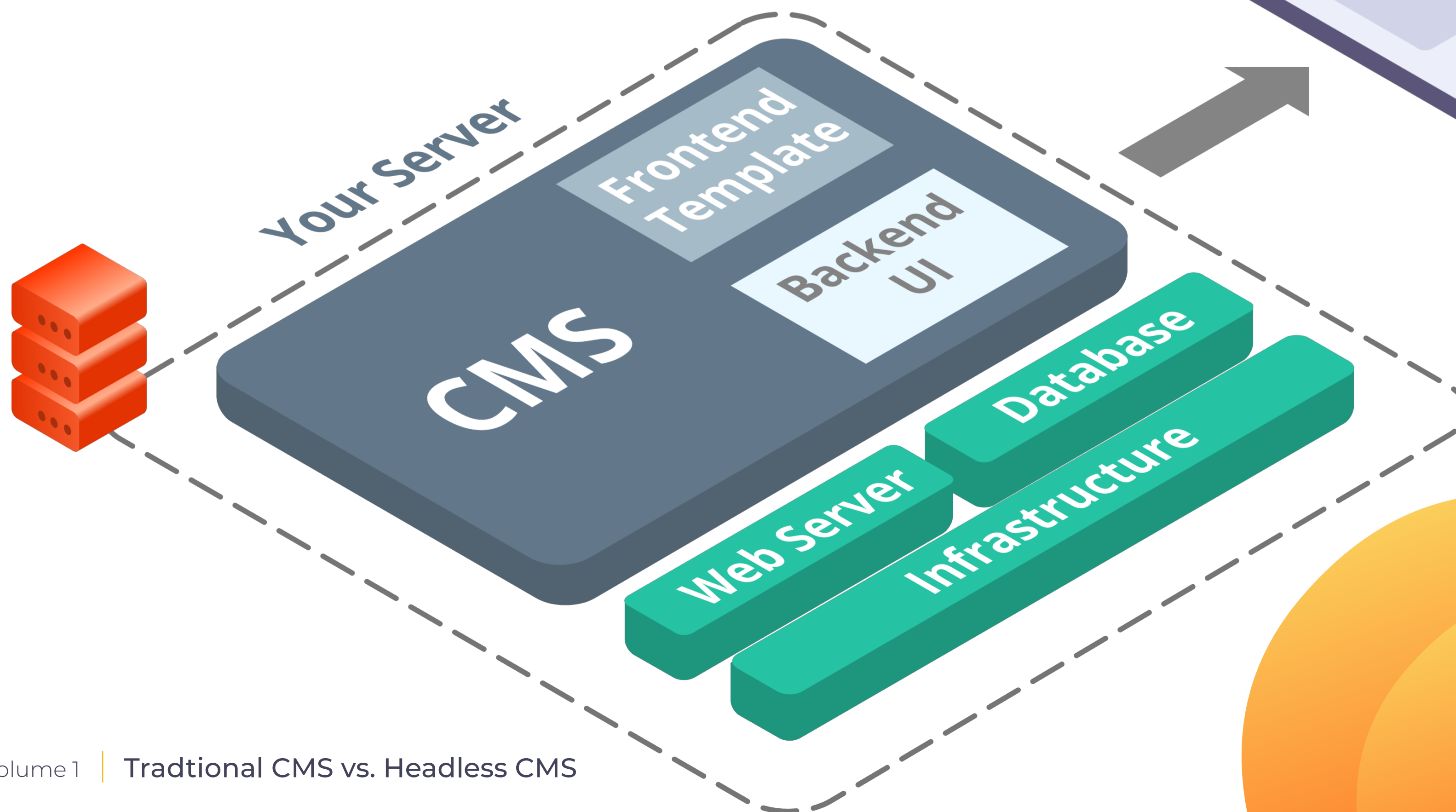
At the very core, a tradtional CMS contains:

- A database for storing data and content

- A backend where content is managed and updated, typically via a WYSIWYG form

- An application to build design templates and schemas

- A built-in frontend which displays published content for users to see on HTML web pages

TRADITIONAL CMS ARCHITECTURE

1. The content management application (CMA) includes the back-end for administering a site and managing content, built into the CMS.

2. A content delivery application (CDA) includes the front-end for delivering and displaying content to the website, built into the CMS.

3. A traditional CMS is a coupled system; the CDM and CDA are built together, so all programming, content, and formatting exist in one inseparable, monolithic system.
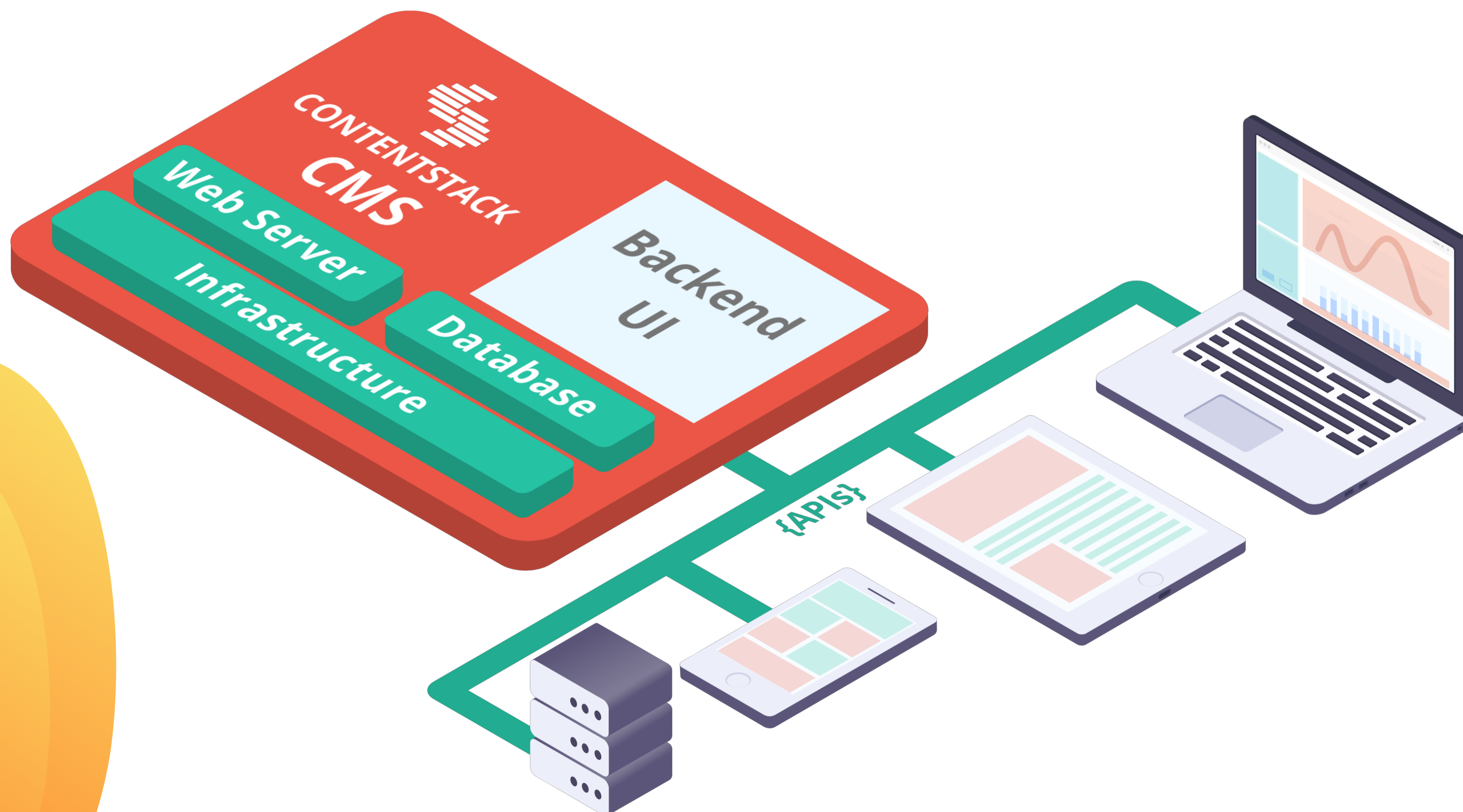
Headless CMS Architecture

A headless CMS architecture does not include a frontend built into the system. The system has administration, content management, and editorial capabilities (which can be presented to the user via very simple, intuitive interfaces) but it allows you to build websites and apps on any frontend of your choice – independent from your content management system. A headless CMS uses APIs or web services to publish the content. In essence, the same content can be distributed to a website, app, smartphone, smartwatch, and the Internet of Things (IoT) because the content is not tied to any predetermined schema or structure.

The headless CMS architecture includes the following:

- A content management backend

- API/Web services support

- A microservices architecture

Headless CMS Architecture Diagram

1. A headless CMS separates or decouples the CMS and CDA environments, so content is separate from programming and formatting.

2. A headless API-first CMS enables omnichannel content delivery, meaning it can deliver content to ANY device via APIs.

3. A headless CMS frees developers to use any framework and is future-proof, allowing developers to use APIs to integrate with new technologies and applications.

NOT JUST FOR DEVELOPERS

A headless CMS is rightly known as a developer-friendly API-first CMS. With Contentstack, developers get all the tools, SDKs, frameworks needed to benefit from the latest technology and build cool sites and apps. However, it also delivers on addressing the pain points of the content manager and business user.

Contentstack is a CMS that offers the best of both worlds; thoughtfully optimized for your content and business managers while also catering directly to the needs of your developers. To expand on that, business users have the control to manage their content without having to wait on IT or an agency. They can publish content themselves; process that is as simple as filling out a web form and hitting the Publish" button and that comes with safeguards built-in.

Besides enabling business users to publish content across all digital channels instantly, there are numerous safeguards in place to prevent them from "breaking" anything. It's a winning combination of self-service with quality control. Contentstack is a powerful and intuitive headless CMS solution that caters equally to developers and marketers with its cleverly designed features and capabilities.

THE CHOICE IS YOURS

Now that we've covered what you need to know about traditional and headless CMSs, you have all the information you need to choose the best solution for your organization. By sitting down with your teams and determining your businesses requirements you can identify critical capabilities and a CMS that best meets your needs. In the end, you'll want to find an easy-to-use, robust solution that has the features and functionality required and is future-proof, so you can quickly adapt to whatever the next technology breakthrough brings.

ABOUT CONTENTSTACK

Contentstack is the hub powering omnichannel content, digital experiences and personalized customer journeys. We are the industry leader in the headless content management system (CMS), delivering a better way for developers and marketers to manage content.

Website: www.contentstack.com
Email: support@contentstack.com

Stay Connected: