

What Is DevContentOps?



Enterprise companies are investing heavily in content-driven applications and digital experiences in an attempt to funnel content into the ever-growing landscape of web, mobile, wearable, and IoT touchpoints. A CMS is crucial to this, but not every CMS platform is ready for the rapid development and deployment of content-driven applications. When it comes to their CMS, **only 16% of marketers said they have the right tech and are using its full potential**¹.

Unfortunately, many CMSs have outdated architectures that are difficult to integrate with DevOps processes, or don't foster collaboration between developers and marketers. For the successful development of content-driven applications, however, companies need a new approach: the DevContentOps™ process.

Let's take a closer look at the drawbacks of a pure DevOps approach when it comes to content-driven apps, and how the expanded DevContentOps process enables a new way of collaboration for content-enabled digital app development.

It Started with DevOps

We've all heard about DevOps, but what does it really mean? In a nutshell, DevOps is a cultural shift that brings together development and IT teams to improve the development, testing, deployment and operations of software. The key elements of a DevOps approach are continuous integration, continuous delivery, configuration as code, and containerization. While these processes have enabled enterprises to build better software, secure it, and get it into production faster, this doesn't mean they should stop there.

Where DevOps Falls Short

Most enterprises that adopt a DevOps culture see substantial improvements in their software development lifecycle, but pure DevOps falls vastly short when it comes to supporting content-enabled apps -- modern multi-channel websites, mobile apps, e-commerce experiences, video platforms, and more. In fact, most software apps have content to some degree. Traditional DevOps says nothing about fostering collaboration between the content authors, software developers, and IT operations. Nor does it suggest the tooling required to integrate a CMS into DevOps processes.

¹ Rose, R. (2019, April 24). The Struggle is Real for Content Management and Technology [New Research]. Retrieved from <https://contentmarketinginstitute.com/2019/04/content-management-technology-research/>

NO SUPPORT FOR CONTENT APPS

While DevOps addresses the challenges of traditional software development, it does nothing to support the content within those applications. Content applications need to have content management separately by marketers, but that doesn't mean content and code need separate repositories. In fact, we'll see later that a single distributed repository is crucial for effectively developing content-driven apps.

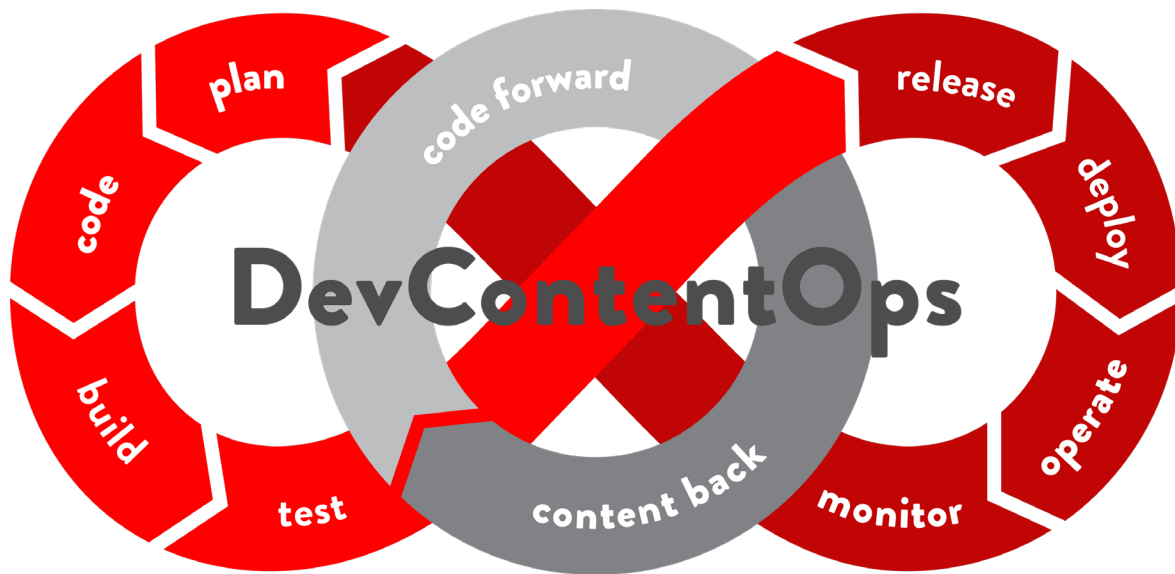
NO COLLABORATION BETWEEN CONTENT MANAGERS AND DEVELOPERS

Content authors and experience designers want new features and design elements to use on their sites and in all other digital experiences. This creates a lot of pressure on development teams to produce at the speed the business requires. In order to deliver, development teams need proper environments with real-world data and content to develop and test in. There is a much closer working relationship between content managers, developers and the operations teams that support them than one might imagine. DevOps is focused on getting new code created and released to production. It does nothing to support the additional creative process and output of the content managers. Further, it doesn't cope well with the fact that developers rely on the output of the content managers to properly create and test their code. The lack of process and technology to support the close-knit relationship between content, code and operations creates process bottlenecks that impact the velocity of all teams involved.

NO INTEGRATION OF CMS TOOLING IN DEVOPS PROCESSES

Chances are it's challenging for your IT operations team to spin up new environments for your CMS. Most companies don't have an automated way to get content in traditional databases from production to lower environments automatically. DevOps alone doesn't prioritize the integration of content management.

Introducing DevContentOps



DevContentOps enables seamless collaboration between developers, IT operations, and content managers. Here are just a few of the benefits of bringing content into the DevOps mix.

SHORTER DEVELOPMENT LIFECYCLE

DevContentOps shortens the lifecycle for delivering features, bug fixes, or other updates to content-based applications with a single data repository. Managing code, configurations, and content in one place enables efficiency in packaging and releasing content apps because manual processes like database replication are no longer necessary. DevContentOps brings content into the CI/CD pipeline.

CONTINUOUS INTEGRATION...AND CONTINUOUS PUBLISHING

A core tenant of DevOps is continuous integration (CI) - or merging code frequently - to avoid substantial merge conflicts. The idea, however, didn't carry over to content management - until now. With DevContentOps, content authors can continuously publish content without being held back by slow release cycles or outages when

new code is being deployed that's common with traditional CMSs. Frequently updating the all-in-one repository also leads to a streamlined develop, test, release, publish cycle for content apps, and less chance of significant merge conflicts.

FRictionLESS DEVELOPMENT

Bringing content into the DevOps process leads to more efficient development and better integration of content with software features. Standard tooling like versioning and configuration management can eliminate situations such as code or content freezes and duplicate publishing of content. A single code and content repository also gives developers the flexibility to use the IDEs, frameworks, or programming languages they choose. They're no longer reliant on the tech stack of a particular CMS. The DevContentOps approach doesn't dictate the development or DevOps tooling possibilities.

SEAMLESS DEVOPS INTEGRATIONS

DevContentOps supports integration with CI/CD services and other DevOps tooling. With traditional CMS platforms, it's harder to implement automated processes because code, scripts, templates, and content are in separate repositories and databases. A DevContentOps approach means developers can still use the tools they're familiar with like Jenkins or Bamboo, but now they can include content in the process as well.

CODE FORWARD, CONTENT BACK

A Git-based CMS enables the seamless migration of content and code amongst different environments, from development to production. Developers can quickly push code from lower environments to testing and production environments using automated tools. Even more importantly, IT can spin up new environments without the hassle of database migration by pulling content back from higher environments. DevContentOps enables simplified content and code migration.

Crafter CMS: Content Meets DevOps

DevContentOps tears down the barrier between development, content creation, and IT operations. The cultural change from DevOps broke down silos, and led to dramatic changes in the way software is developed and deployed. With DevContentOps, there's a new cultural shift that will transform how content is published, and how companies deliver dynamic digital experiences to audiences.

The key to DevContentOps is a single repository for code, configuration, and content. Crafter is built on a Git-based repository that makes developing, deploying, and managing content-driven applications straightforward. Furthermore, the CMS has completely decoupled authoring and delivery repositories, so content changes aren't made in the production environment. Rather, they're pushed from the authoring repository to the live repository when ready using automated processes. Crafter leverages Git to handle everything from versioning and content locking to migrating content between environments. DevContentOps is the crucial next step in adapting to the demands of the digital age.

Getting Started

LOOKING TO LEARN MORE?

- ◇ Video: [Introducing Crafter CMS 3.1](#)
- ◇ Learn about the straightforward developer workflow: [Developer workflow](#)
- ◇ Download Crafter CMS: [Download](#)
- ◇ Launch an AWS Instance of Crafter CMS: [Launch AMI](#)
- ◇ Request a Trial: [Contact us](#)

ABOUT CRAFTER SOFTWARE

Crafter Software is on a mission to replace the broken paradigm of traditional content management, and to usher in a new era of fast, agile and easier development of innovative digital experiences. Our flagship product, Crafter CMS, is amazing for developers, easy for content authors, and fantastic for DevOps. We build our software solutions on the foundation of open source, transparency, robust architecture, high performance, superior quality and outstanding customer support. Available on premise in the enterprise or SaaS in the cloud.

Learn more at craftersoftware.com and craftercms.org.

LEARN MORE

DevContentOps is a trademark of Crafter Software Corporation