

# From Static to Dynamic: Next.js Rendering Patterns with Drupal CMS.

---

## Abstract

Next.js's new App Router introduces powerful rendering strategies like Server-Side Rendering (SSR), Static Site Generation (SSG), and Incremental Static Regeneration (ISR). But how do these techniques apply when you're using Drupal as a headless CMS?

This session explores how to effectively pair the Next.js App Router with Drupal as your content backend. We'll walk through how to fetch Drupal content via JSON:API or GraphQL and map each rendering strategy to real-world use cases—marketing pages, blogs, landing pages, and dynamic user content. You'll leave with a clear understanding of how to choose the right rendering approach based on content freshness, performance, and complexity.

---

## Session Summary

This session explains how to integrate Drupal with Next.js using the new App Router and make smart choices between SSR, SSG, ISR, and dynamic rendering based on your content needs. Learn how to fetch content via JSON:API or GraphQL, implement revalidation strategies, and deliver fast, SEO-friendly experiences.

---

## Session Synopsis

**From Static to Dynamic: Next.js Rendering Patterns with Drupal CMS.**

Modern frontend frameworks give developers exciting new tools—but also new decisions. With the arrival of Next.js App Router, we now have multiple rendering options: Server-Side Rendering (SSR), Static Site Generation (SSG), and Incremental Static Regeneration (ISR). But how do these strategies map to a backend like Drupal, with its rich editorial workflows and flexible content model?

In this session, we'll explore how to effectively use Drupal as a headless CMS with the Next.js App Router. We'll walk through how to connect to Drupal via JSON:API or GraphQL, and use the right rendering strategy for different use cases.

### What You'll Learn:

- The practical differences between SSR, SSG, ISR, and dynamic rendering in the App Router
- How to integrate Drupal content using the **Next-Drupal** client.
- When and how to use functions like `generateStaticParams`, `generateMetadata`, and `revalidatePath`
- Tips for building SEO-friendly decoupled sites.
- Performance tuning and architecture patterns for decoupled apps.

We'll wrap it up with a **live demo** that shows a single Drupal node rendered three different ways, plus a breakdown of trade-offs in performance, freshness, and complexity.

Whether you're a Drupal developer curious about headless frontends, or a Next.js fan looking to work with Drupal content, this talk will equip you with the tools and understanding you need to build smarter, faster, and more flexible web experiences.

---

## Key Takeaways

- Understand the differences between SSR, SSG, ISR, dynamic rendering and Partial Pre-rendering in Next.js App Router.
  - Learn how to fetch and display Drupal content using JSON:API.
  - Know when and how to revalidate content for freshness.
  - Discover performance tuning tips and decoupled best practices.
- 

## Target Audience

- Frontend developers using or exploring Next.js with Drupal
  - Full-stack developers building decoupled apps
  - Drupal developers curious about modern frontend rendering strategies
- 

## Technical Level

**Intermediate** (Attendees should be familiar with either Drupal or Next.js basics)

---

## Format

### **40-minute session**

- 25 minutes: presentation + live code walkthrough
- 10 minutes: demo (live rendering comparison)
- 5 minutes: Q&A

---

## Optional Add-ons

- Live comparison: SSR vs SSG vs ISR using a single Drupal node
- Code snippets for `generateStaticParams`, `generateMetadata`, and `revalidatePath`
- Visual architecture diagram: Pages Router vs App Router with Drupal.