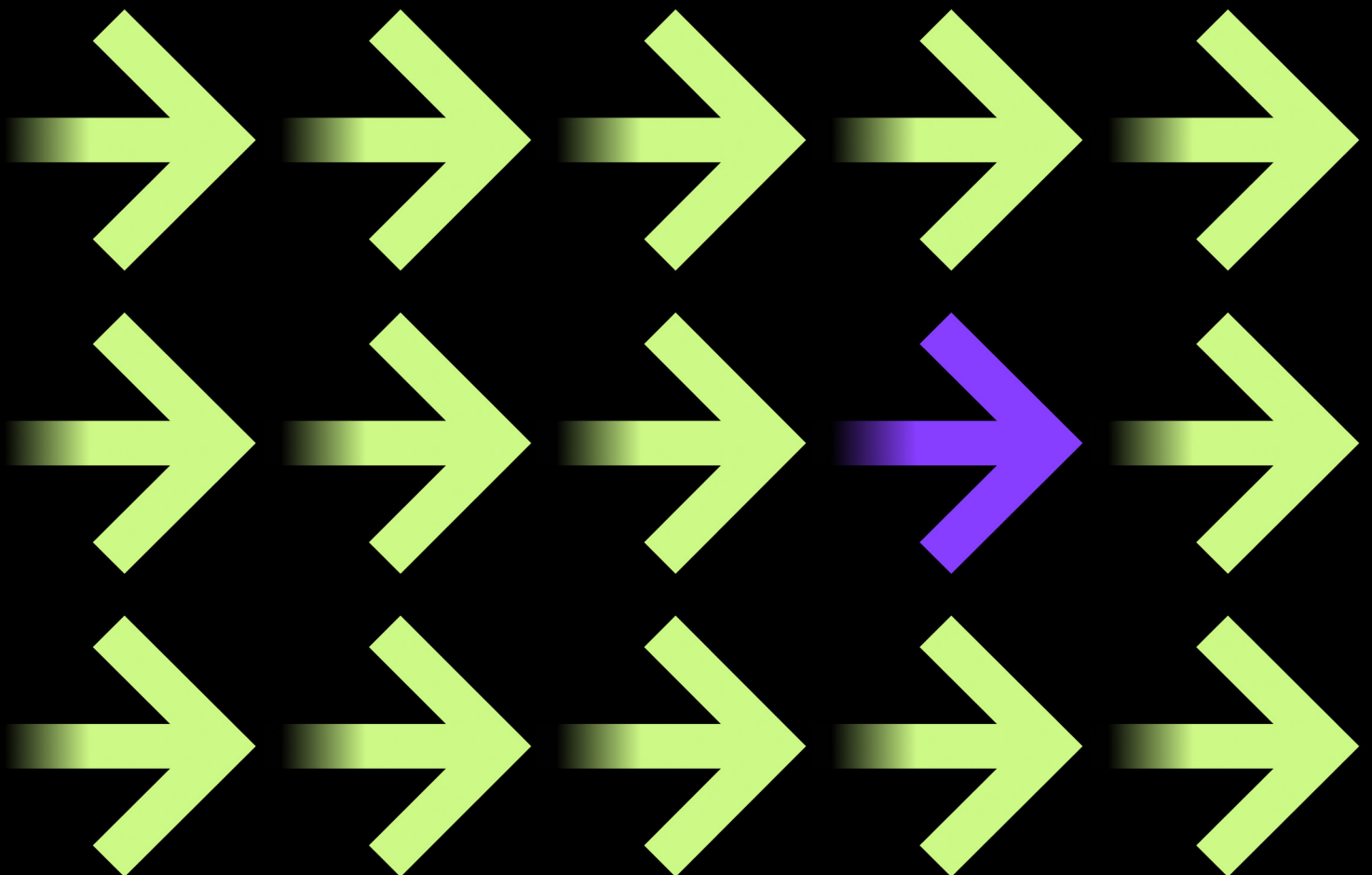


WooCommerce to Shopify

A proven migration framework for
technical decision makers moving
from WooCommerce to Shopify.



Contents

Executive Summary	3
Key considerations when migrating from WooCommerce	4
Understanding the shift from an open-source platform	5
Hosting and maintenance	7
Security and compliance	9
Site performance and load speeds	10
Customizing Shopify	11
Platform extensibility	16
Performance and security	19
Data migration	20
The migration process with DigitalJAS	28
Additional considerations	36
DigitalJAS and Shopify set you up for long-term success	39
What's next?	41

Executive summary

The changing ecommerce landscape

The ecommerce industry has evolved significantly in recent years, with increasing demands for performance, scalability, and seamless user experiences. As businesses grow, their technology stacks must keep pace with rising customer expectations and operational complexity.

Many companies initially choose WooCommerce for its open-source flexibility and lower initial cost (more to come on the higher management costs later), allowing them to develop highly customized online stores. However, as they scale, they often encounter challenges such as performance bottlenecks, security vulnerabilities, costly maintenance, and limited scalability and stability.

Shopify is built to deliver what technical leaders need most: a secure, scalable, and customizable commerce platform that eliminates these common pain points. The platform combines fully hosted infrastructure, global CDN-powered performance, built-in PCI-DSS compliance and unlimited scalability—all while reducing total cost of ownership. Whether you're concerned about customization flexibility, data migration integrity, or maintaining business continuity, Shopify's architecture and tools are designed to address these critical technical considerations.

In this guide, technical decision makers will get all the information they need to navigate a migration from WooCommerce to Shopify. We'll explore key considerations across hosting and maintenance, security and compliance, customization, data migration, and more. We'll also review a typical migration process to help illustrate what the journey looks like—and why DigitalJAS is the best partner for the job.

About DigitalJAS

DigitalJAS is a Shopify Plus Partner agency with 20 years of ecommerce experience. We specialize in guiding mid-market and enterprise brands through platform strategy, Shopify optimization, and complex migrations. Our mission is simple: eliminate complexity, unlock performance, and help merchants grow faster.

Key considerations when migrating from WooCommerce

Technical leaders evaluating a migration from WooCommerce to Shopify face several critical decisions that will impact their business's long-term success. Here, we'll explore the fundamental considerations—from understanding the shift from open source to SaaS to evaluating security, performance, and customization capabilities. We'll also examine how Shopify addresses these common technical concerns while highlighting the strategic advantages of a managed commerce platform.

Consideration

01

Understanding the shift from an open-source platform

Migrating from WooCommerce to Shopify is more than just a platform switch—it's a fundamental shift in how an ecommerce business operates. To ensure a smooth transition, it's essential to understand the differences between a SaaS-based solution (Shopify) and an open-source platform (WooCommerce).

Shopify

Shopify reduces or even eliminates many operational concerns by offering:

- **A fully hosted environment:** Shopify manages uptime, speed optimization, and security.
- **Automatic updates and maintenance:** No need to manually update plugins or core software.
- **Native security and compliance:** Shopify handles PCI compliance, fraud prevention, and secure payments.
- **Seamless scalability:** Shopify's cloud infrastructure supports businesses of all sizes, from startups to enterprise brands.

WooCommerce

WooCommerce is a popular choice because of its open-source nature, offering extensive customization. However, this flexibility also means businesses must handle:

- **Hosting and server management:** Selecting and maintaining hosting providers, ensuring server uptime, and managing site performance.
- **Security and compliance:** Implementing PCI compliance, SSL certificates, and fraud protection measures.
- **Plugin and code updates:** Constant monitoring of third-party plugins and code updates to avoid compatibility issues.
- **Scaling challenges:** High traffic can cause slowdowns, requiring businesses to invest in dedicated hosting or content delivery networks (CDNs).

While WooCommerce allows for deep customization, these responsibilities can become a bottleneck for growing businesses.

By shifting from WooCommerce to Shopify, CTOs free up technical resources to focus on innovation rather than maintenance.

Shopify eliminates many operational concerns



Consideration

02

Hosting and maintenance

One of the most immediate differences you'll notice is in hosting and infrastructure management. This operational shift impacts everything from your team's daily workflows to your long-term technical strategy.

Shopify

With Shopify, hosting is bundled and fully managed, giving you:

- 24/7 infrastructure support
- 99.9% uptime and automatic scaling
- No plugin updates or server tuning required

This managed approach allows your developers and engineers to focus on innovation—not maintenance.

WooCommerce

Hosting costs and site upkeep add up quickly on WooCommerce. Whether you're using SiteGround, WP Engine, or a custom solution, you're likely dealing with:

- Manual server configuration
- Scheduled backups and restoration planning
- Plugin conflicts after updates
- Uptime monitoring and debugging

Business continuity and disaster recovery

Data backup and recovery reveal another critical operational difference—one that directly impacts business continuity and risk management.

Shopify

- Built on a highly redundant cloud infrastructure that ensures data safety
- Automatic data storage mitigates the risk of data loss
- Rollback functionality requires third-party apps, providing enhanced data management options

WooCommerce

- Requires third-party backup plugins or managed hosting plans with recovery options
- Backup timing and completeness are your responsibility
- Limited failover support without advanced hosting setup

Shopify minimizes the risk of total failure or unrecoverable data. In the event of a technical failure, Shopify’s team handles mitigation and resolution—something WooCommerce stores must plan for independently.

Summary: Why Shopify meets the CTO’s checklist

CTO concern	Shopify	WooCommerce
Performance	Fast, global CDN with built-in optimization	Hosting-dependent, variable
Security	PCI-DSS compliant and fully managed	Must manage patches, SSL, PCI
Hosting	Included and optimized	DIY or third party
Compliance	Built-in tools (GDPR, fraud, PCI)	Manual implementation
Maintenance	Drastically reduced maintenance requirements	Requires regular developer time
TCO	Higher entry, lower long-term cost	Low entry, high hidden costs
Scalability	Auto-scaling for growth and peak traffic	Manual tuning and optimization

Consideration

03

Security and compliance

When evaluating ecommerce platforms, security and compliance are critical considerations. WooCommerce and Shopify take different approaches to managing these responsibilities, with Shopify automating key security measures as part of the platform, while WooCommerce requires a more hands-on approach from your team.

Shopify

As a Level 1 PCI-DSS compliant platform, Shopify handles security automatically with:

- SSL certificates across all pages
- Built-in fraud analysis and detection
- Real-time threat monitoring
- Managed security patches and updates
- Data encryption and access control
- GDPR compliance tools for customer data management

WooCommerce

As an open-source WordPress plugin, WooCommerce places the security burden on your team. This includes:

- Manual configuration of SSL certificates
- Self-managed PCI-DSS compliance implementation
- Regular plugin and theme updates to prevent vulnerabilities
- Third-party tools for backups and malware scanning
- Ongoing security patch management

Automated security features enable IT teams to focus on growth-oriented tasks rather than ongoing maintenance. This allows for more efficient use of technical resources.

Consideration

04

Site performance and load speeds

Ensuring fast and reliable site performance is crucial for maintaining a positive user experience and maximizing conversions. Shopify and WooCommerce handle performance differently, with Shopify including performance-driven hosting within a SaaS model, with continued support and optimization. Meanwhile, WooCommerce requires manual optimization and hosting management.

Shopify

Shopify removes the guesswork by offering enterprise-grade performance out of the box with a fully managed infrastructure:

- Built-in global CDN powered by Fastly for worldwide low latency
- Optimized servers managing everything from TTFB to asset compression
- Automatic scaling for flash sales and traffic spikes data management

WooCommerce

WooCommerce's performance depends entirely on how well your hosting environment is configured. If you're using shared hosting or low-tier VPS servers, performance can suffer significantly during peak traffic periods. Page load times, time to first byte (TTFB), and server response times are all variable—and often require investment in external optimization tools. To maintain acceptable performance, businesses must manage:

Built-in global CDN powered by Fastly for worldwide low latency

- Hosting environment and monitoring
- Caching solutions (e.g., WP Rocket, W3 Total Cache)
- Third-party CDN configuration (e.g., Cloudflare)
- Plugin performance impact
- Server capacity during peak traffic periods

Real-world impact: Faster page loads directly improve conversion rates. According to Cloudflare, even a one-second improvement in load time can increase conversions by up to 2%. Shopify helps you avoid that risk without manual tuning.

Consideration

05

Customizing Shopify

MYTH:

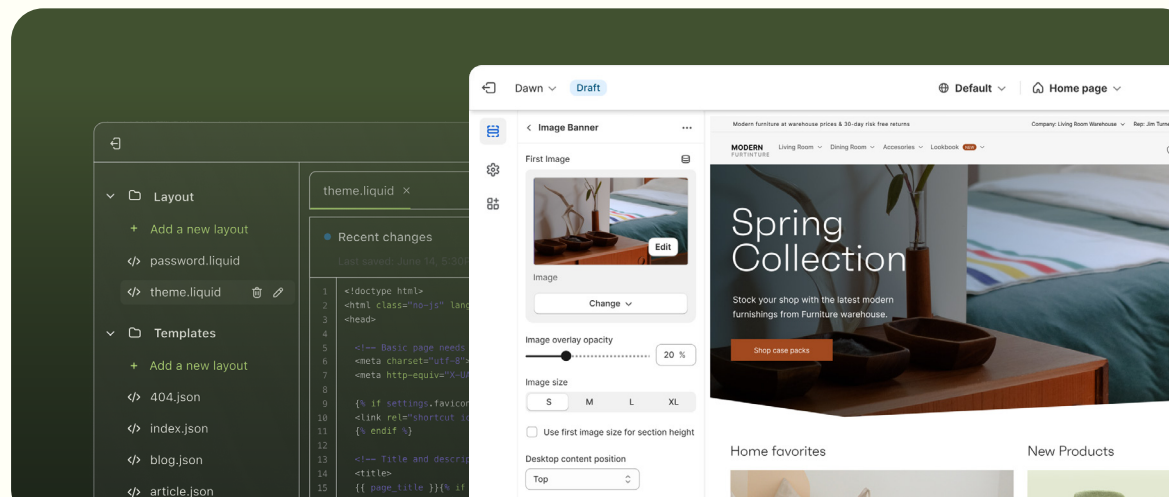
Shopify is too restrictive compared to WooCommerce

REALITY:

Shopify offers extensive customization while maintaining stability

A common misconception among CTOs and development teams is that Shopify's SaaS model limits customization compared to WooCommerce. In reality, Shopify provides powerful tools that allow businesses to tailor their store experience while benefiting from a stable, managed infrastructure.

This section explores Shopify's key customization capabilities, debunking the notion that it is restrictive.



Balancing customization with stability

For businesses that rely on deep customizations in WooCommerce, Shopify offers an extensible framework that balances flexibility with long-term stability. Here's a high-level comparison of how the two platforms approach flexibility and customization:

Feature	Shopify	WooCommerce
Themes and UI customization	Liquid-based themes with customizable sections and blocks allowing for dynamic content	Custom themes via PHP and WordPress templating
Product and data modeling	Metaobjects and metafields for advanced data customization	Custom post types and fields
Checkout experience	Checkout extensibility and Shopify Functions for tailored checkout flows	Highly customizable, but requires heavy development with strong risk of instability
Integrations and APIs	Robust API ecosystem, app marketplace, and webhook support	Open-source plugins and API connections
Performance optimization	Drastically reduced maintenance requirements	Requires caching, CDNs, and server tuning

Shopify's structured approach ensures businesses can achieve the same level of customization as WooCommerce while benefiting from:

- **Liquid templating and Storefront API:** Enabling custom storefront designs
- **Metafields and metaobjects:** Extending Shopify's data structure without plugins
- **Custom checkout experiences:** Using Shopify Functions for tailored checkout flows
- **Public and custom apps:** Developing custom apps that integrate seamlessly with Shopify APIs

Theme customization: Liquid, sections, and theme API

Shopify's approach reduces complexity while maintaining high flexibility, making it easier to scale and update without breaking functionality.

WooCommerce vs. Shopify theme development comparison

Feature	Shopify	WooCommerce
Templating language	Liquid templating language	PHP and WordPress templating
Customization approach	Modular and flexible with reusable sections and blocks	Full code control but dependent on WordPress structure
Cost	Some free themes, one-time cost for others	Some free themes, ongoing annual cost for others
Page builders	Built-in drag-and-drop editor with Online Store 2.0	Requires third-party plugins like Elementor
Codebase complexity	Maintains structured codebase with regular updates	Often requires multiple plugins and custom functions

Why Shopify's theme system is more scalable:

- **Liquid templating language:** Allows developers to create dynamic, data-driven themes with structured control
- **Sections everywhere:** Modular editing system enabling businesses to customize layouts without coding
- **Theme API and custom blocks:** Developers can extend themes with dynamic components for personalized experiences.

Checkout customization: Checkout extensibility and Shopify Functions

Shopify now offers deep checkout customization through:

- **Checkout extensibility:** Businesses can add custom fields, branding, and logic without modifying core checkout code.
- **Shopify Functions:** Developers can inject custom logic for discounts, payment rules, and checkout flows. Functions are hosted by Shopify, helping with performance and scalability at no additional cost.
- **One-page checkout:** Optimized for higher conversion rates

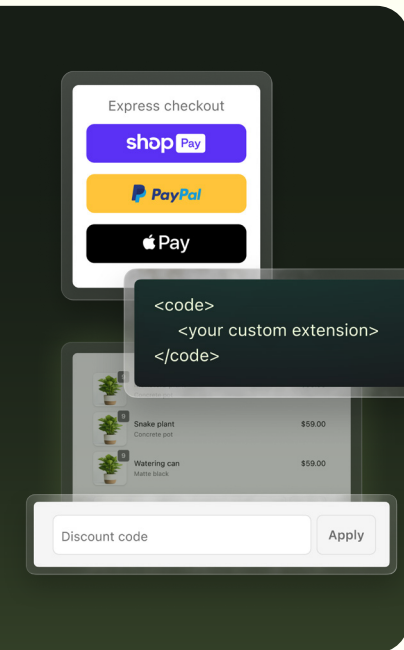
Example use cases:

1. **Custom shipping logic:** Display unique delivery options based on ZIP code.
2. **Tiered discounts:** Apply advanced promotions based on order value or customer type.
3. **B2B checkout customization:** Restrict payment methods for wholesale accounts.

Constraints of Shopify Checkout customization:

Shopify has fine-tuned their checkout over billions of transactions. As a result, there are some constraints that brands should be aware of when planning migration. Generally, these exist to ensure an optimized checkout flow and performance while keeping you on Shopify's upgrade path.

1. **Limited UI control:** UI extensions can only be placed in predefined checkout areas (e.g., after shipping, before payment).
2. **Limited styling control:** Styling is done via the Branding API or settings, not custom CSS.
3. **Third-party app compatibility:** Not all third-party apps are built to work with checkout extensibility.



In short, Shopify provides structured customization within a secure environment. Meanwhile, WooCommerce does allow checkout modifications, but it often requires heavy coding or unreliable plugins, resulting in unstable experiences.



CASE STUDY

How V's Barbershop improved efficiency in site updates using Shopify's customization tools

V's Barbershop, a premium men's barbershop with over 60 locations across the US, was operating their website front end on Wordpress and WooCommerce. They were looking for a more robust CMS where brand content, appointment scheduling, and ecommerce functionality could seamlessly work together.

The brand faced challenges with limited front-end control, reliance on third-party tools for UX, and weak integration between brand, store, and content. In migrating to Shopify, they adopted mobile-first Liquid themes with easy-to-manage templates, leveraged metaobject-driven content for brand storytelling, and integrated blogs and products directly into page sections for a unified experience.

Results:

- Site speed improved by over 30%.
- Site conversion (appointments booked, products sold) increased by over 40%.
- Site management and update efficiency increased drastically.

Shopify provides the best of both worlds: extensive customization without the headaches of WooCommerce's maintenance and content limitations.

Consideration

06

Platform extensibility

Public and custom apps via Shopify APIs

One notable difference between WooCommerce and Shopify is how each platform extends functionality through third-party tools. Shopify uses apps, while WooCommerce uses plugins. Both serve similar purposes—enhancing the core platform—but they differ in how they’re distributed, maintained, and integrated.

Types of Shopify apps

App Type	Purpose	Example use cases
Public apps (listed)	Available on Shopify App Store; built for use and compatibility with multiple stores	Subscription billing, SEO optimization, marketing automation
Public apps (unlisted)	Developed by agencies for multiple clients	Custom ERP connectors and Checkout functionality
Custom apps	Built for a single store’s internal use	Custom warehouse integrations, proprietary pricing rules

What are Shopify Plus Certified Apps?

The Shopify Plus Certified App Program supports the largest Shopify businesses by helping them find the apps and solutions they need to build and scale their business. Learn more about the certification program and requirements [here](#).

Shopify apps vs WooCommerce plugins comparison

Aspect	Shopify Apps	WooCommerce plugins
Availability	Centralized in the Shopify App Store, with curated listings	Wide range available from multiple sources including WordPress.org, marketplaces, and direct developer sites
Installation	One-click installation via the Shopify App Store	Installed via WordPress admin or manual upload
Review and approval process	Reviewed by Shopify before publishing: both for the App Store and to earn the “Built for Shopify” badge	No formal review process; quality varies
Updates and maintenance	Regular updates managed through the App Store framework	Handled by individual developers; timing and support can vary
Performance considerations	Apps run in a more controlled environment with limited impact on storefront performance	May impact site performance depending on code quality
Theme compatibility	Built to work within Shopify’s architecture, including app blocks	Potential for conflicts with custom themes or other plugins
Support structure	App developers are required to offer timely support along with clear methods of contact	Developer-dependent
Pricing models	Generally monthly subscriptions; most offer free plans	Often free or one-time fee; some with renewals
Customization	Customization options depend on app APIs	Highly customizable for technical users
Platform compliance	Apps must meet Shopify’s platform and compliance standards	Responsibility lies with the plugin provider

Why Shopify's app model is more secure:

- Strict app store guidelines: Ensures quality, security, and performance standards
- API rate limits and webhooks: Optimized for stable integrations without overloading servers
- No plugin conflicts: Apps work within Shopify's framework, reducing site breakage risks.

For businesses migrating from WooCommerce, Shopify's app ecosystem provides safer, more efficient ways to extend store functionality.

Integrations and extensibility: APIs, webhooks, and automation

Seamless third-party integrations

Shopify's robust API system allows businesses to connect essential tools such as:

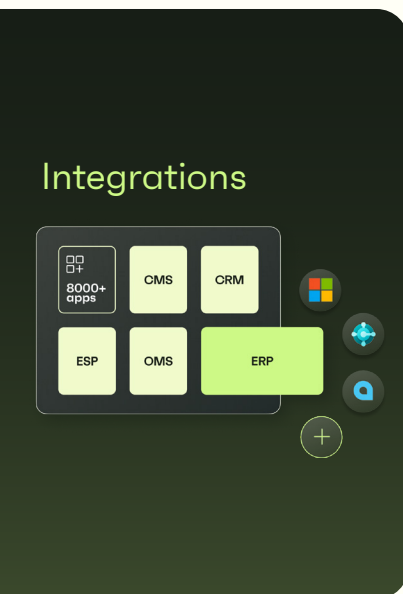
- ERP and inventory systems: NetSuite, SAP, Brightpearl
- Marketing and CRM tools: Klaviyo, HubSpot, Salesforce
- Shipping and logistics: ShipStation, EasyPost, 3PL integrations

Webhooks and event-driven automations

Shopify supports real-time data syncing through:

- **Webhook triggers:** Automate tasks like order confirmations and inventory updates.
- **Shopify Flow and automations:** No-code automation for customer segmentation and workflows.
- **GraphQL APIs:** Faster, more flexible data access compared to WooCommerce's REST-only API

These capabilities ensure that businesses can fully integrate Shopify into their existing tech stack.



Consideration

07

Performance and security

With Shopify, performance and security aspects are handled by Shopify through its managed cloud infrastructure, while on WooCommerce they depend on your hosting setup and ongoing maintenance.

Concern	Shopify	WooCommerce
Hosting	Fully hosted with global CDN	Self-managed, requires optimization
Performance	Optimized servers with automatic scaling	Relies on caching and third-party optimizations
Security	PCI-DSS compliant out of the box with built-in fraud prevention	Requires manual SSL setup, PCI compliance, and fraud protection
Updates and maintenance	Automatic updates, no maintenance required	Manual updates, risk of plugin conflicts

Why this matters:

- **Faster load times:** Shopify's infrastructure ensures better speed, improving SEO and conversion rates, according to a [recent report](#).
- **No maintenance hassles:** No need for constant security patches or plugin compatibility checks
- **Enterprise-grade security:** Shopify handles key aspects of compliance like PCI, data encryption, infrastructure security, fraud monitoring, audits, and more, so businesses don't have to.

Consideration

08

Data migration

MYTH:

Migrating from WooCommerce to Shopify is complex and risky

REALITY:

With the right tools and process, it's seamless and reliable

Data integrity and migration concerns

Data migration is often a concern when moving from WooCommerce to Shopify, but with the right approach, businesses can ensure:

- **Accurate product mapping:** Retaining SKUs, inventory levels, and product variations
- **Seamless order and customer history transfer:** Migrating past transactions without data loss
- **SEO preservation:** Implementing proper URL redirects and metadata transfers

One of the most common concerns CTOs raise when considering a migration is the potential risk of data loss, inconsistencies, or downtime. Shopify and WooCommerce have different data architectures, which can seem intimidating. But with the right approach, migrating to Shopify can be not only safe—but also an opportunity to clean, optimize, and enhance your data.

Mapping WooCommerce’s data model to Shopify’s

Although both platforms manage similar core data—products, customers, orders, content—the way they store and structure it differs significantly. Here’s a breakdown of how data maps during a typical migration:

WooCommerce	Shopify
Products (post type)	Products with variants and options
Categories and tags	Collections and product tags
Orders	Orders (including timeline and fulfillment history)
Customers	Customers, customer tags, and customer metafields
Coupons and discounts	Discount codes and Shopify Functions
Media (library)	Product/asset files (uploaded to CDN)
Pages/posts	Shopify pages and blogs
Custom fields	Metafields and metaobjects (products, collections, customers, orders, pages, etc.)

Key differences to consider:

- Shopify simplifies product organization by providing a streamlined system for options and variants, making it easier to restructure complex WooCommerce setups.
- Downloadable files, bundles, and subscriptions need careful review to match Shopify's model or available apps.
- URLs and permalinks need redirects to maintain SEO rankings (covered in testing/launch section).

Using automation and APIs to transfer data

Manual data migration is time-consuming and error-prone. That's why we recommend using specialized tools combined with Shopify APIs to automate the process.

Recommended tools and strategies:

1. Matrixify (formerly Excelify)

- Industry-standard tool for structured import and export of products, customers, orders, pages, and more
- Supports bulk updates and field mapping pulled from WooCommerce APIs
- Allows for staging uploads and rollback options

2. Shopify API

- Enables custom scripts for migrating nuanced data or connecting with ERP/CRM systems & WooCommerce REST APIs
- Useful for transforming data mid-transfer (e.g., formatting product specs into metafields)

3. Scripting in Python/Node.js

- Ideal for custom migrations or pre-processing large datasets

 **shopify** storefront API

Ensuring historical data integrity

Maintaining customer order history, tags, and segmentation is vital for marketing and retention. During migration, we:

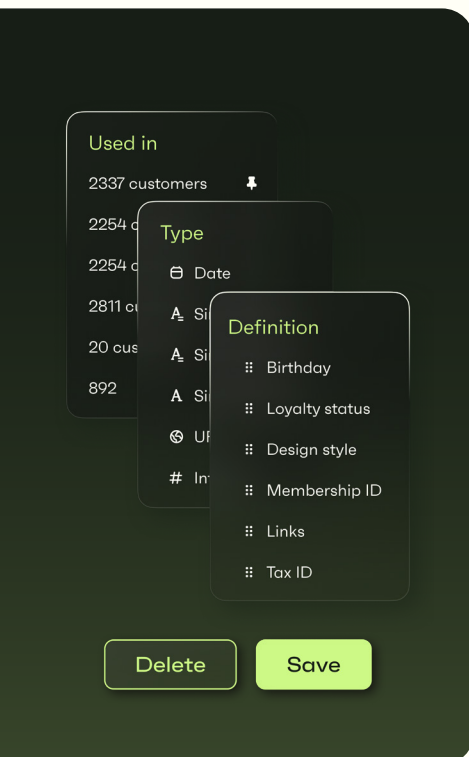
- Match WooCommerce order data to Shopify's format (including customer association).
- Transfer fulfillment status and notes into Shopify's timeline.
- Recreate customer profiles with complete order history and lifetime value.
- Retain customer groups via Shopify's tags, segments, and metafields.
- Implement URL redirects and metadata transfers.

All data is verified through dry runs and tested in staging environments to ensure no records are lost or duplicated.

Advanced data modeling: Metaobjects, metafields, and dynamic content

Shopify simplifies advanced data processes with:

- **Metafields:** Extend products, customers, orders, and pages with custom attributes.
- **Metaobjects:** Define structured, reusable data models for complex content needs.
- **Dynamic content APIs:** Enable personalized storefronts based on customer behavior or product attributes.



Example use cases:

1. **Dynamic content:** Custom fields for product and category (collection) data offer a dynamic experience within single templates that allow for better product content and filtering.
2. **Advanced product specifications:** Store detailed tech specs in metaobjects for filtering and searchability.
3. **Personalized content blocks:** Serve dynamic content (e.g., recommended products) based on browsing history.

By using Shopify's structured data system, businesses can enhance customer experiences without relying on third-party plugins or custom code like they would with WooCommerce, which stores additional product and customer data using custom post types, fields, and taxonomies.

Enhancing your data during migration

Migrating isn't just about transferring data—it's a chance to enhance it:

- **Enrich product pages:** Add structured specifications, cross-sells, and brand content via metafields or metaobjects.
- **Clean and consolidate customer lists:** Remove duplicates, normalize naming and email formats, and apply tags for segmentation.
- **Standardize SEO metadata:** Migrate page titles, meta descriptions, and custom canonical tags cleanly to preserve rankings.
- **Improve inventory accuracy:** Reconcile inventory across channels and sync with a Shopify-compatible inventory management system.



CASE STUDY

French Dressing Jeans

French Dressing Jeans (FDJ), a Canadian fashion brand, had built their online business through WooCommerce with thousands of SKUs and over five years of customer and order data. They feared that migrating would mean losing vital history or disrupting ongoing fulfillment.

The challenges included managing over 1,000 products with variants, custom attributes, and legacy data; handling orders spanning many years with various statuses and notes; and consolidating customer records split between guest checkouts and registered accounts.

The migration strategy:

- Used Matrixify integration to WooCommerce API to bring in default product data, and migrate existing custom attributes to new Shopify metafields for multi-purpose use (PDP dynamic content, product filters, collection creation)
- Migrated full customer list and order history including reference to old SKU naming conventions for full reporting functionality
- De-duplicated customer list and created dynamic customer groups via tags
- 100% data transfer — no records lost

The results:

- Improved product filtering and merchandising on the front end
- Enhanced customer segmentation for email and loyalty campaigns
- Full historical accuracy for reporting and planning

Data migration best practices and checklist

Pre-migration:

- ☐ Conduct a data audit (products, customers, orders, SEO metadata).
- ☐ Identify and clean duplicate or obsolete data.
- ☐ Review custom fields or attributes needing metafield equivalents.
- ☐ Back up WooCommerce data.

During migration:

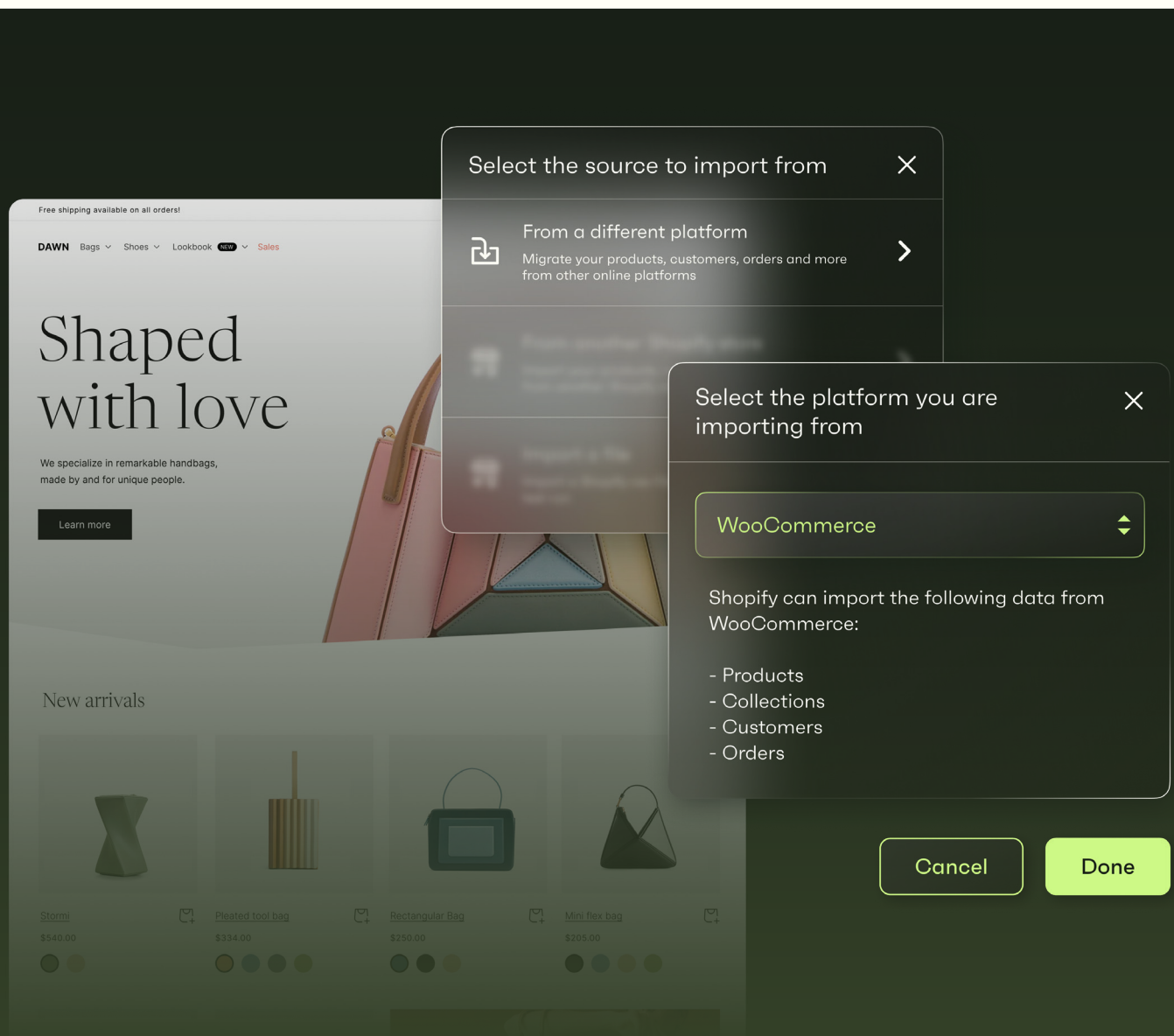
- ☐ Consider using a staging environment for test imports.
- ☐ Validate mappings (SKU formats, image links, taxonomy structures).
- ☐ Monitor for failed records or API rate limits.
- ☐ Perform dry runs before importing live data.
- ☐ Pause any inventory integrations, marketing automations and/or other systems that may be triggered due to new orders.

Post-migration:

- ☐ Review imported data for completeness.
- ☐ Perform order simulations and test checkout flows.
- ☐ Ensure historical order timelines and notes were retained.
- ☐ Review analytics and CRM integrations for continuity.

During migration:

- Consider using a staging environment for test imports.
- Validate mappings (SKU formats, image links, taxonomy structures).
- Monitor for failed records or API rate limits.
- Perform dry runs before importing live data.
- Pause any inventory integrations, marketing automations and/or other systems that may be triggered due to new orders.



The migration process with DigitalJAS

DigitalJAS has guided dozens of businesses through platform migrations—from lean startups to enterprise retailers. A successful WooCommerce to Shopify migration requires more than just data transfer. It demands strategic planning, deep platform knowledge, and post-launch support to ensure a smooth, high-performance result.

Their five-phase process is designed to help your store smoothly transition to Shopify, aiming to avoid downtime and establish an architecture built with the future in mind.

STEP 1:

Discovery and audit

OBJECTIVE:

Understand the current WooCommerce architecture, business requirements, and pain points.

DigitalJAS begins every migration with a comprehensive audit of your WooCommerce setup, team workflows, and technology stack.

WHAT THEY ASSESS:

- Product catalog structure (variations, bundles, downloadable products, custom fields)
- Order history, customer segmentation, and sales data
- Third-party plugins, integrations, and custom code
- Current performance, security risks, and hosting setup
- SEO structure (URLs, metadata, redirect mapping)

DELIVERABLES:

- Technical audit summary
- Recommended migration path
- Initial timeline estimate
- Key risks and mitigation plan

BONUS:

DigitalJAS often uncovers hidden performance issues or redundant processes that can be streamlined post-migration.

STEP 2:

Data migration strategy

OBJECTIVE:

Design a structured plan for migrating products, customers, orders, and custom fields with 100% data integrity.

This is the most critical step of the process. The team doesn't just copy and paste data—they rearchitect it to work natively within Shopify's system.

WHAT THEY DO:

- Map WooCommerce product fields to Shopify products, variants, and metafields

- Consolidate customer records (handling duplicates, guest checkouts)
- Convert WooCommerce order history to Shopify orders with correct associations
- Translate categories and tags to Shopify collections and smart collections
- Build URL redirect map to preserve SEO and backlinks

KEY TOOLS:

- Matrixify for structured data import/export
- Shopify admin and GraphQL APIs for advanced logic
- Python or Node.js scripts for transformations (e.g., bundle to variant models)

STEP 3:

Store build and customization

OBJECTIVE:

Design and configure your new Shopify store to reflect your brand, UX requirements, and business goals.

Migrating is a unique opportunity to modernize your front end, optimize for conversion, and clean up legacy clutter.

WHAT'S INCLUDED:

- Custom Shopify theme (using Liquid and Online Store 2.0 features)
- Page designs using drag-and-drop sections and blocks
- Custom UX elements (sticky nav, dynamic filters, shoppable content)
- Shopify app selection and integration
- Custom checkout logic using Shopify Functions

DigitalJAS replicates core functionality from your WooCommerce store—then enhances it using Shopify-native best practices.

STEP 4:

Testing and validation

OBJECTIVE:

Ensure the migrated store is fully functional, accurate, and ready for real users.

Testing goes beyond clicking through a few pages. The DigitalJAS team runs QA across multiple layers of your new store.

TEST CASES:

1. Product variants display and pricing logic
2. Promotions, discounts, and custom checkout flows
3. Data integrity for migrated customers and orders
4. Email templates and transactional communications
5. Responsive design and cross-browser performance
6. Redirect map and SEO configuration

Testing goes beyond clicking through a few pages. The DigitalJAS team runs QA across multiple layers of your new store.

STEP 5:

Launch and optimization

OBJECTIVE:

Launch your new Shopify store with no disruption to customers—and optimize continuously.

Prelaunch checklist:

- ☐ Final data sync for orders/customers placed during staging
- ☐ Domain switch and DNS propagation
- ☐ Payment gateway and fulfillment system final validation
- ☐ Go-live rollback plan (just in case—but rarely used)

POST-LAUNCH SUPPORT:

- 30–60 days of hands-on support for training, app tuning, and bug fixes
- Shopify analytics and GA4 event tracking setup
- Performance monitoring and speed optimization
- Ongoing consultation for CRO and retention strategies

Why the DigitalJAS process works

Migration phase	Key outcome
Discovery	Clear plan, no surprises
Data strategy	Clean, accurate, SEO-safe data
Build	A store built for speed and scale
Testing	Confidence before launch
Optimization	Continued growth and improvements

The DigitalJAS process is designed for minimal business disruption, maximum data integrity, and high post-migration performance.

Estimated migration timelines

While every migration is different, we do see certain patterns emerge, and can predict what factors affect migration time. Here are some high-level estimates of migration times based on various factors:

Complexity	Factors	Example use cases
Low/typical	• Small product catalog (under 500 SKUs) • Standard theme requirements • Minimal custom functionality • Simple content and page structure	~8-10 weeks
Medium	• Mid-sized catalog (500–5,000 SKUs) • Some third-party integrations • Moderate custom design or functionality • Multiple content types/pages to migrate	~12-16 weeks
High	• Large/complex catalog (5,000+ SKUs, variants, custom attributes) • Heavily customized UX or workflows • Complex integrations (ERP, OMS, subscriptions, etc.) • Migration from multiple systems or platforms	~24+ weeks



CASE STUDY

Anthony's Espresso

Anthony's Espresso is a retailer with a long history of selling and servicing high-end espresso machines and accessories, operating both a successful online store and multiple retail locations across Canada. For years, they relied on WooCommerce for eCommerce and a separate third-party POS system for in-store sales. Their primary frustrations included:

- Inventory discrepancies between retail and online channels
- Disconnected customer profiles, with no unified purchase history or link between POS and Ecommerce.
- Difficulty managing promotions across POS and eCommerce
- Rising maintenance costs, performance issues, and fragmented data
- Limited front end experience allowing them to scale the online business.

Anthony's team wanted a centralized solution that could streamline operations, unify their commerce experience, and scale with their growing business.

Migration to Shopify Plus and POS:

- Detailed Discovery phase to run data modeling, functional planning, and omni-channel commerce strategy to ensure that launch included a more unified experience to support both product sales and product service.
- Full data migration of products (including reassigned and newly created product attributes), customer, and order history (including both purchase orders as well as service orders.)
- Implementation of an updated, mobile first website with BOPIS.
- End to end product training for both the onsite and in-store teams.

Results:

- Team efficiency improved by 50%.
- Fully unified customer experience where customers can buy online, pick up in store, or view the status of an in-store service invoice on their online account.
- Large website performance improvement including conversion and revenue increases of over 50% in the first year

In today's retail landscape, having both physical and online stores is a tremendous opportunity to meet your customers where they want to shop. Utilizing Shopify's advanced tooling in both channels, as well as the unified commerce functionality to vastly improve the customer experience, will only help ensure a company capitalizes on that opportunity.

Additional considerations

In addition to the content already covered in this document, there are other potential features that should be considered to ensure a seamless transition and continued functionality. These features can significantly impact the user experience and back-end operations, so careful planning is essential.

These features may include, but are not limited to:

Onsite search:

Evaluate your current on-site search solution. If using a third-party search tool compatible with both platforms, migration can be straightforward. Alternatively, consider switching to a Shopify-compatible solution or utilizing Shopify's native search functionality to minimize technical debt.

Loyalty programs:

Loyalty and rewards programs play a critical role in customer retention. Assess whether your existing loyalty program provider integrates with Shopify. In some cases, migrating to a third-party provider well-supported on the platform may be preferable. Very often these platforms are set up to migrate programs and historical data between them. Plus, utilizing Shopify Flow can help in the link between Shopify and your loyalty program.

Subscription services:

Subscriptions are increasingly popular for recurring revenue. If your existing subscription provider works with both WooCommerce and Shopify, migration may be simple. If not, explore Shopify's native subscription features or trusted third-party apps to maintain continuity.

Product reviews:

Migrating customer reviews is essential to maintain social proof. If using a third-party review app that works on both platforms, migration is often straightforward. Otherwise, explore Shopify's compatible reviews partners to retain and display customer feedback effectively.

Omnichannel commerce (aka, unified commerce):

Shopify offers native unified commerce where online store, POS, inventory, and customer data operate from a single platform. This integration powers real-time inventory syncing across locations, unified customer profiles with complete order history, and seamless

Shopify's
unified model



omnichannel capabilities. Shopify also provides built-in marketplace integrations without requiring separate plugins. What's more, multi-location inventory management, smart fulfillment routing, and comprehensive reporting are included natively, reducing the need for additional inventory management solutions. Consider how your existing omnichannel operations will map to Shopify's unified ecosystem, prioritizing native functionality to reduce integration complexity and maintenance. All of this stands in contrast to WooCommerce's patchwork approach that requires multiple plugins and platforms.

Here's a visual snapshot comparing the unified commerce capabilities of both platforms:

Feature	Shopify	WooCommerce
Online store	Native hosted	WordPress-based
POS	Native POS with sync	Third-party integrations
Inventory	Unified, multi-location	Plugins
Customer profiles	Unified, real-time	Fragmented
Channel selling	Built-in integrations	Separate plugins
Fulfillment and routing	Smart routing engine	Manual or plugin-based
Analytics	Unified, real-time	Plugin and GA
Maintenance	Auto-updates included	Ongoing dev and plugins

In each case, it is vital to assess the Shopify ecosystem for native or well-supported alternatives. This approach helps reduce technical debt, simplifies maintenance, and ensures the longevity of your ecommerce operations.

DigitalJAS and Shopify set you up for long-term success

Migrating from WooCommerce to Shopify is a bold and strategic decision—and the right one for businesses looking to eliminate technical debt, improve performance, and scale their business faster.

As we've outlined in this playbook, Shopify provides:

- A stable, high-performance infrastructure with built-in scalability
- Powerful customization through themes, APIs, metafields, and Shopify Functions
- A structured, secure approach to data migration with full historical integrity
- Predictable total cost of ownership with lower long-term maintenance
- A rich ecosystem of apps and integrations for every stage of growth

Key takeaways for CTOs:

Consideration	Shopify solution
Customization limitations	Liquid, storefront API, Shopify Functions, public or custom apps
Data migration complexity	Structured tools (Matrixify, APIs), automated processes
Performance bottlenecks	Global CDN, optimized infrastructure, scaling on demand
Security risk	PCI-DSS compliance, built-in SSL, managed hosting
Plugin maintenance	App ecosystem with no version conflicts or manual updates
Cost unpredictability	Fixed platform fees, reduced DevOps costs
Managing a complicated tech stack	Unified commerce platform with a shared central database

Shopify doesn't just replicate your WooCommerce setup—it elevates it. CTOs gain confidence knowing their infrastructure is future-proof, and their teams are empowered to focus on growth and innovation rather than firefighting and updates.

What's next?

If you're exploring a move from WooCommerce to Shopify, here are a few recommended steps to get started:

Schedule a discovery session:

Our team will walk through your current architecture, identify opportunities for optimization, and create a migration roadmap tailored to your business.

Request a data audit:

We'll assess your catalog, customers, and order history to evaluate the migration scope and flag any risks upfront.

Explore Shopify with a pilot:

Spin up a staging store and let us replicate your key business logic, so you can see how Shopify can handle your unique requirements.

If you're ready to learn more about migrating from WooCommerce to Shopify, get in touch today.

CONTACT SHOPIFY

And if you're ready to get started today with a partner who knows what it takes, let's get to work.

DIGITALJAS

Sources

[“Built for Shopify Requirements,”](#) Shopify.

[“Checklist of Requirements for Apps in the Shopify App Store,”](#) Shopify.

[“Commerce Moves Faster on Shopify,”](#) Shopify.

[“How Website Performance Affects Conversion Rates,”](#) Cloudflare.