

Executive Summary

Migrations of commerce platforms are complicated projects. They demand careful preparation. This preparation involves setting clear goals, creating a complete strategy for moving data and protecting SEO, mapping integrations, performing extensive testing and more.

The decision to migrate your commerce platform is one of the most significant strategic choices a business can make. It's a high-stakes endeavor that can either unlock tremendous growth or lead to costly setbacks. By adopting a modern approach and leveraging the right architecture, you can transform this complex undertaking from a high-risk gamble into a highly strategic move that future-proofs your business.

Migration Approaches

When it's time to move to a new commerce platform, you'll need to decide how to get there. There are two primary approaches, each with its own set of trade-offs.

- **Big Bang:** This is a high-risk, all-at-once switch. It's faster and simpler. However, there's no way to go back to the old system. This approach works best for small businesses with basic requirements.
- **Phased:** A gradual move. It has a lower risk, allowing for continuous testing and less downtime. This method is perfect for large, complex businesses. It can be broken down further into several sub-strategies, including functional, geographic, and customer-based phasing, along with the "strangler fig pattern."

A Better Approach: Composable Commerce

A modern architectural style, composable commerce is a better option for migrations. It's built on MACH principles (Microservices, API-first, Cloud-native, Headless). Composable commerce lets businesses use a "best-of-breed" approach. They can pick individual, modular parts and connect them with APIs. This setup supports phased migrations, decreases reliance on a single vendor, and provides unmatched flexibility and scalability.

Modern Migration Considerations

For modern migrations, you must also consider **agentic commerce as part of your new platform**. AI-powered agents, like [KIBO's Shopper Agent or Merchandiser Agent](#), help to automate and personalize tasks, driving greater productivity. A composable, API-first architecture is vital for bringing in these agents. It provides long-term advantages, like lower operational costs and better efficiency.



This guide will walk you through the key considerations, strategies, and modern technologies needed to execute a successful commerce migration.

Planning a Commerce Migration

A successful migration isn't about luck; it's about meticulous planning. The work you do before a single line of code is moved will be the most critical part of the entire process. This pre-migration assessment and planning phase is crucial because it sets the foundation for a successful transition and helps to prevent major issues later on.

Key Elements of Pre-Migration Planning

The planning phase of a commerce migration is comprehensive and involves many moving parts. Here's a breakdown of the key elements you must consider:

1 Defining Goals and Objectives:

Before you do anything else, you need to clearly articulate why you're migrating. Is it for better performance, new features, or to reduce costs? By defining your goals, you can choose the right platform and measure the project's success. Common goals include improving site speed, enhancing user experience (UX), and increasing scalability to handle more traffic and products.

2 Data Migration Strategy

Your data is the lifeblood of your commerce store. A successful migration depends on transferring all critical data accurately and securely from your old platform to the new one. This includes:

- **Customer data:** profiles, logins, order history, contact information and loyalty data.
- **Product data:** descriptions, images, pricing, categories and inventory.

- **Order data:** historical records, payment and returns history and shipping details.
- **Content:** blog posts, static pages, search settings, SEO data, reviews and other marketing materials.
- **Subscriptions:** active continuity orders and historical data.
- **Marketing and Promotions:** Discount Codes & Coupons, Email Subscriber Lists.
- **Fulfillment History:** Details on which location fulfilled the order, and the shipping information.

You should have a plan for data cleansing to remove duplicates or outdated entries and a backup strategy in case something goes wrong.

3 SEO Preservation

A poorly executed migration can tank your search engine rankings and organic traffic. You must have a robust plan to maintain your search engine optimization (SEO) value. This includes:

- **Redirects:** Setting up 301 redirects for all old URLs to point to their new locations.
- **URL mapping:** Creating a comprehensive list of old URLs and their corresponding new URLs.

- **Monitoring:** Keeping a close eye on your site's performance in search engines before and after the migration.

4 Integrations and Customizations

Modern ecommerce platforms rely on integrations with third-party systems like payment gateways, customer relationship management (CRM) tools, and enterprise resource planning (ERP) systems. You need to audit your current tech stack and ensure that all existing integrations are compatible with the new platform. If they're not, you'll need to plan for new integrations or custom development work.

A strategic approach is to partner with a platform that offers pre-built integrations, like the KIBO Connect Hub. This kind of solution provides a significant advantage by allowing you to rapidly scale your business with minimal custom development. Instead of building complex point-to-point connections, the Connect Hub leverages a unified API approach, which masks complex systems behind a single API endpoint. This not only accelerates the integration process but also simplifies ongoing maintenance.

With the KIBO Connect Hub, you can seamlessly connect to a wide range of systems through pre-built integrations, including trading partners, payment adapters, shipping carriers, ERPs, CRMs, marketplaces, and more.

This framework is designed to reduce the burden on your development teams, allowing you to quickly deliver new capabilities to the market by leveraging a battle-tested integration platform.

The architecture is built on a Cloud Native Message Bus and uses serverless functions, enabling workflows to respond to events from KIBO or a third-party platform. This focus on event-driven orchestration and a unified API approach provides the speed and flexibility needed for a successful modern migration.

5 User Experience and Functionality Testing

The new site must provide a seamless shopping experience for your customers. Thorough testing is non-negotiable. This involves:

- **Functional testing:** Checking that all features, like the search function, shopping cart, and checkout process, work as expected.
- **User acceptance testing (UAT):** Having key stakeholders and a group of users test the site to ensure it meets business needs and user expectations.
- **Performance testing:** Making sure the site can handle expected traffic volumes without slowing down or crashing.

6 Stakeholder Alignment and Communication

A migration is a company-wide effort, not just an IT project. All key stakeholders, including marketing, sales, and customer service teams, must be involved and informed. A clear communication plan helps manage expectations and ensures everyone is aligned on the project goals and timeline.

Types of Commerce Migrations

When you're ready to migrate your commerce platform, choosing the right approach is a major determinant of your project's success. Let's dive into the two most common strategies and explore when each makes the most sense for your business: **Big Bang Migration** and **Phased Migration**.

Big Bang Migration

This approach involves a single, all-at-once switch from your old commerce platform to the new one. The entire migration of data, systems, and processes is completed in a very short, pre-defined window, often over a weekend or during a period of low traffic. When the new system goes live, the old one is completely shut down.

Best for:

Businesses with tight deadlines and desire, or when the legacy system has critical, unfixable flaws that require an immediate transition to address technology gaps and meet goals.

Advantages:

- **Faster Deployment:** The new site is launched quickly, allowing you to realize the benefits of the new platform sooner.
- **Lower Initial Costs:** You avoid the resources and costs associated with maintaining and synchronizing two systems in parallel.
- **Simplicity:** The process is a single, focused event, which can be easier to manage from a project coordination standpoint.

Disadvantages:

- **High Risk:** A single failure can be catastrophic, leading to a complete system outage and significant data loss. There is no fallback to the old system if something goes wrong.
- **Limited Testing:** Real-world testing with live traffic and user behavior is not possible before the final launch.
- **Intense Pressure:** The short timeline and high stakes place immense pressure on the migration team to execute a flawless plan.

Phased Migration

Also known as a trickle migration or staged migration, this approach involves a gradual transition. The old and new platforms run simultaneously for a period, and different components—such as customer data, products, or specific site functionalities—are migrated in a series of manageable phases.

Best for:

Complex, high-traffic ecommerce businesses with lots of data and many integrations. It's ideal for companies that need to launch quickly, can't risk downtime, and prefer a phased rollout.

Advantages:

- **Lower Risk:** By moving in stages, you can identify and resolve issues as they arise without impacting the entire system.
- **Reduced Downtime:** Business operations are not interrupted, as the old system remains active until the new one is fully functional and stable.
- **Enhanced Testing:** Each phase can be thoroughly tested with real-world users, allowing for continuous refinement and improvement.

Disadvantages:

- **Increased Complexity:** Managing and synchronizing two live systems at once is technically complex and needs careful planning to prevent data inconsistencies.
- **Higher Costs:** You may incur higher costs for a longer period due to the maintenance and licensing of both the old and new platforms.
- **Extended Timeline:** The entire migration process takes significantly longer to complete, delaying the full realization of the new platform's benefits.

Flavors of Phased Migration

A phased migration is a strategic approach that offers a way to modernize your platform without the high stakes of an all-at-once switch. However, this isn't a one-size-fits-all solution; it's a flexible strategy with several distinct "flavors" that can be tailored to your specific business needs and risk tolerance. Understanding these different approaches is key to building a migration plan that delivers continuous value while minimizing disruption.

Functional Phasing

This approach involves migrating one specific business function at a time. It's often used when a business has a complex, monolithic platform and wants to replace individual components without overhauling the entire system at once.

Pros:

- **Reduced Risk:** Issues can be isolated to a single function, making them easier to manage and fix without affecting the entire site.
- **Continuous Improvement:** You can deliver new functionality and features to users as soon as a phase is complete, providing a steady stream of value.
- **Specialized Expertise:** This approach allows teams to focus on one function at a time, leveraging specialized expertise for each phase.

Cons:

- **Increased Complexity:** The old and new systems must coexist and be carefully synchronized, which can be technically challenging and requires robust data mapping.
- **Longer Project Timeline:** The overall project takes longer to complete due to the staggered nature of the releases.

Geographic Phasing

This strategy involves migrating your ecommerce platform one geographic region or country at a time. For international businesses, this is a low-risk way to test a new platform's viability and localize your operations step by step.

Pros:

- **Lower Risk:** A failure in one market will not affect your business in other regions, protecting your primary revenue streams.
- **Controlled Testing:** You can test the new platform's localization, payment gateways, and shipping integrations in a smaller, less critical market before a global rollout.
- **Scalable Lessons:** You can apply the lessons learned from the initial launch to subsequent migrations in other markets.

Cons:

- **Management Overhead:** It requires the management of multiple teams and staggered timelines, which can be administratively complex.
- **Data Consistency:** Maintaining data consistency across different instances of the platform for various regions can be challenging.

Customer-Based Phasing

Customer-based phasing segments your customer base and migrates them to the new platform in waves. It's a great method for managing risk and gathering early, valuable feedback from a controlled group of users before a full-scale launch.

Pros:

- **Reduced Customer Impact:** Any issues that arise will only affect a small portion of your customer base.
- **Early Feedback:** You can gather valuable feedback from a specific user segment and make improvements before a full-scale launch.
- **Better User Experience:** By addressing feedback in real time, you can ensure a much smoother and more positive experience for the majority of your users.

Cons:

- **Technical Complexity:** It requires a technically sophisticated system to manage and segment different customer groups and route them to the correct platform.
- **Data Synchronization:** You must ensure that all data (e.g., customer history, loyalty points) is accurately synced across both platforms for customers who haven't been migrated yet.

Strangler Fig Pattern

The Strangler Fig Pattern is a common phased migration approach that involves gradually replacing a monolithic system's functionality with new microservices. Named after the strangler fig vine that grows around a host tree, this strategy allows the new system to "strangle" and eventually replace the old one without a high-risk, all-at-once migration.

Pros:

- **Reduced Risk:** It's an incremental, low-risk approach. You can test and refine new features in a live environment without impacting the entire site. If a new service fails, you can simply revert back to the old one.
- **Continuous Improvement:** You can deliver new value to customers as you build each service, rather than waiting for a complete replatforming project to finish.
- **Incremental Investment:** You can spread the cost and effort of the migration over time, focusing on one part of the system at a time.

Cons:

- **Increased Complexity:** You must manage and maintain both the old and new systems simultaneously, which can lead to complex routing logic and integration challenges.
- **Extended Timeline:** The migration process takes a long time to complete because you are replacing the system piece by piece.
- **Potential for Bottlenecks:** The old system's core dependencies can become a bottleneck if not properly managed during the transition.

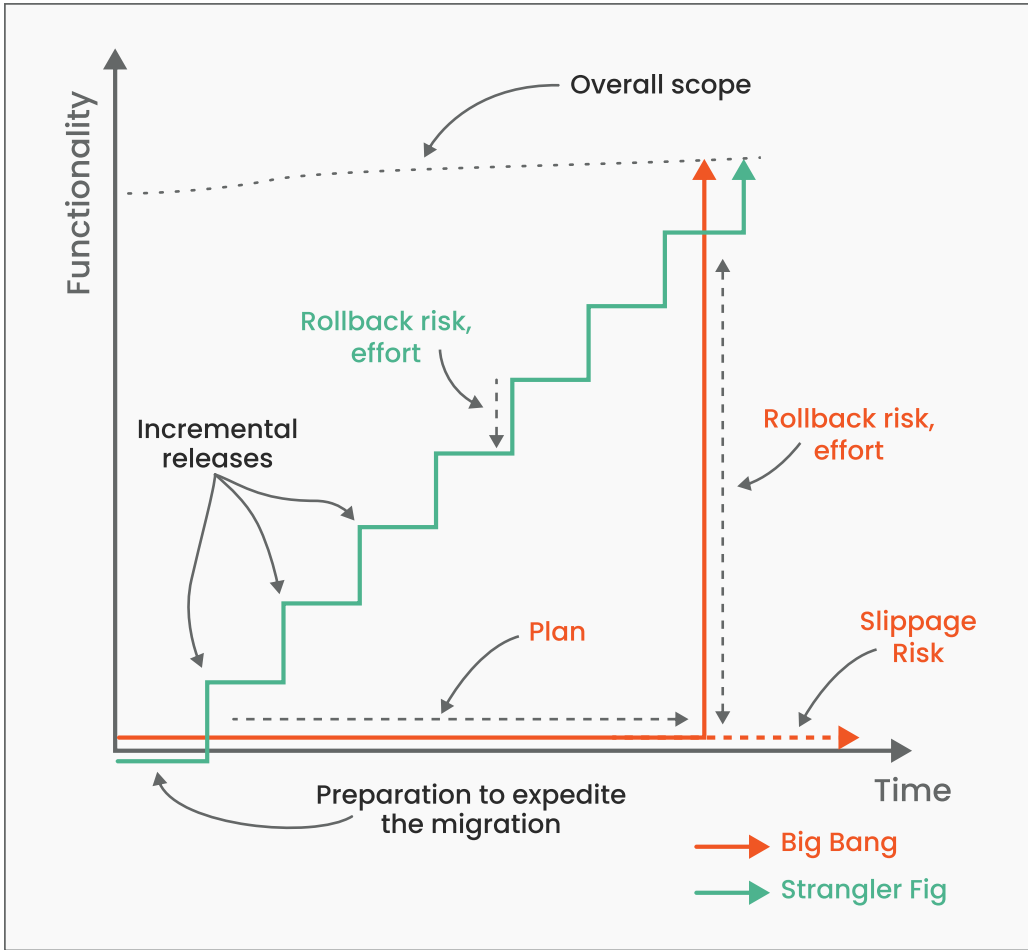


Fig 1. The above image is an illustrative example of steps taken with a Big Bang approach vs. the Strangler Approach. Big Bang and Strangler approaches illustrate different paths to digital transformation. A Big Bang approach involves a long implementation cycle to build a complete new system, followed by a single, high-risk launch that replaces the old system entirely. In contrast, the Strangler approach allows for a phased rollout, where new functionalities are introduced incrementally to replace parts of the legacy platform over time. This reduces risk but also means the transition to the new platform is piecemeal rather than a single, sweeping change. Diagram sourced from thoughtworks.com

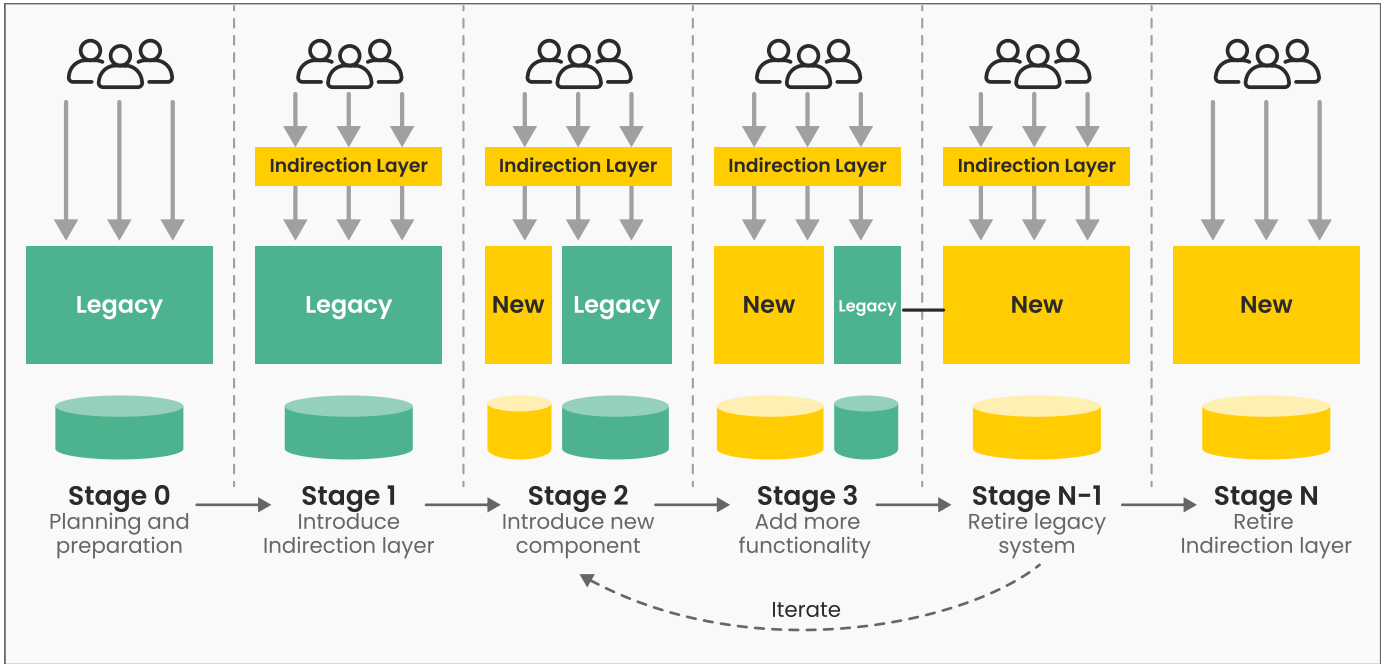


Fig 2. The Strangler approach offers a methodical path for platform migration by incrementally replacing your legacy system. By introducing an Interaction Layer, you can stabilize the process, gradually shifting specific functionalities to a new composable platform. This method allows for continuous deployment and testing in a live environment, providing the opportunity to refine features and optimize performance as you go, without the need to build an entire new system before launch. Diagram sourced from thoughtworks.com

Analytical Comparison

Approach	Pros	Cons	Best for
Functional	Reduced risk, continuous improvement, specialized focus	Increased complexity, longer timeline	Businesses with a monolithic architecture that want to replace components incrementally.
Geographic	Lower risk, controlled testing, scalable lessons learned	Management overhead, potential data inconsistencies	International businesses with operations in multiple regions.
Customer-Based	Reduced customer impact, early user feedback, better UX	High technical complexity, data synchronization challenges	Businesses with a large, segmented customer base that want to test functionality.
Strangler Fig Pattern	Reduced risk, continuous delivery, incremental investment	Increased complexity, long timeline, potential bottlenecks	Companies with a monolithic legacy system that is too risky or expensive to replace all at once.

Transforming Commerce Migrations with Composability

Composable commerce represents a paradigm shift in how businesses build and manage their digital presence, moving away from rigid, all-in-one platforms toward a flexible, modular architecture that puts you in complete control. This fundamental change transforms a typical ecommerce migration from a high-risk, all-or-nothing event into a strategic, low-risk process. By enabling businesses to select and integrate best-of-breed components as needed, [composable commerce](#) vendors, like KIBO, provide the agility and control necessary to modernize your technology stack on your own terms.

Composable Commerce: A Lego Set for Ecommerce

Imagine building a Lego model. A traditional monolithic platform is like a pre-packaged Lego set with a specific design and a fixed set of bricks. You can build that one model, but if you want to create something different or swap out a piece, it's difficult and often requires you to start over.

Composable commerce, on the other hand, is like having a giant bucket of Lego bricks. Each brick is a specialized, high-quality piece (e.g., a specific type of wheel, a unique character, a special roof tile). You can pick and choose the exact pieces you need to build your ideal solution, and if you want to change one part, you simply swap that one brick out without affecting the rest of the structure.



Key Principles of Composable Commerce

This modular approach is built on several key principles, often referred to by the acronym MACH:

- **Microservices:** Instead of a single, massive application, the commerce platform is broken down into a collection of small, independent services. Each service handles a specific business function (e.g., a service for product catalog, another for search, one for promotions, and so on). This allows for individual development, deployment, and scaling of each service.
- **API-First:** All of these microservices communicate with each other and with external systems through Application Programming Interfaces (APIs). This "API-first" approach ensures that the different components are not tied to a specific technology or programming language, making them highly flexible and interoperable.
- **Cloud-Native:** Composable commerce solutions are built and deployed on the cloud, leveraging the power of cloud computing for infinite scalability, high availability, and real-time updates. This allows businesses to handle traffic spikes and growth without managing their own servers.
- **Headless:** This is a crucial concept. "Headless" means decoupling the front-end (the customer-facing interface, or "head") from the back-end (the commerce logic). With a headless setup, you can use any front-end technology to build a website, mobile app, smart kiosk, or any other customer touchpoint, all while using the same back-end commerce engine.

The Evolution of eCommerce

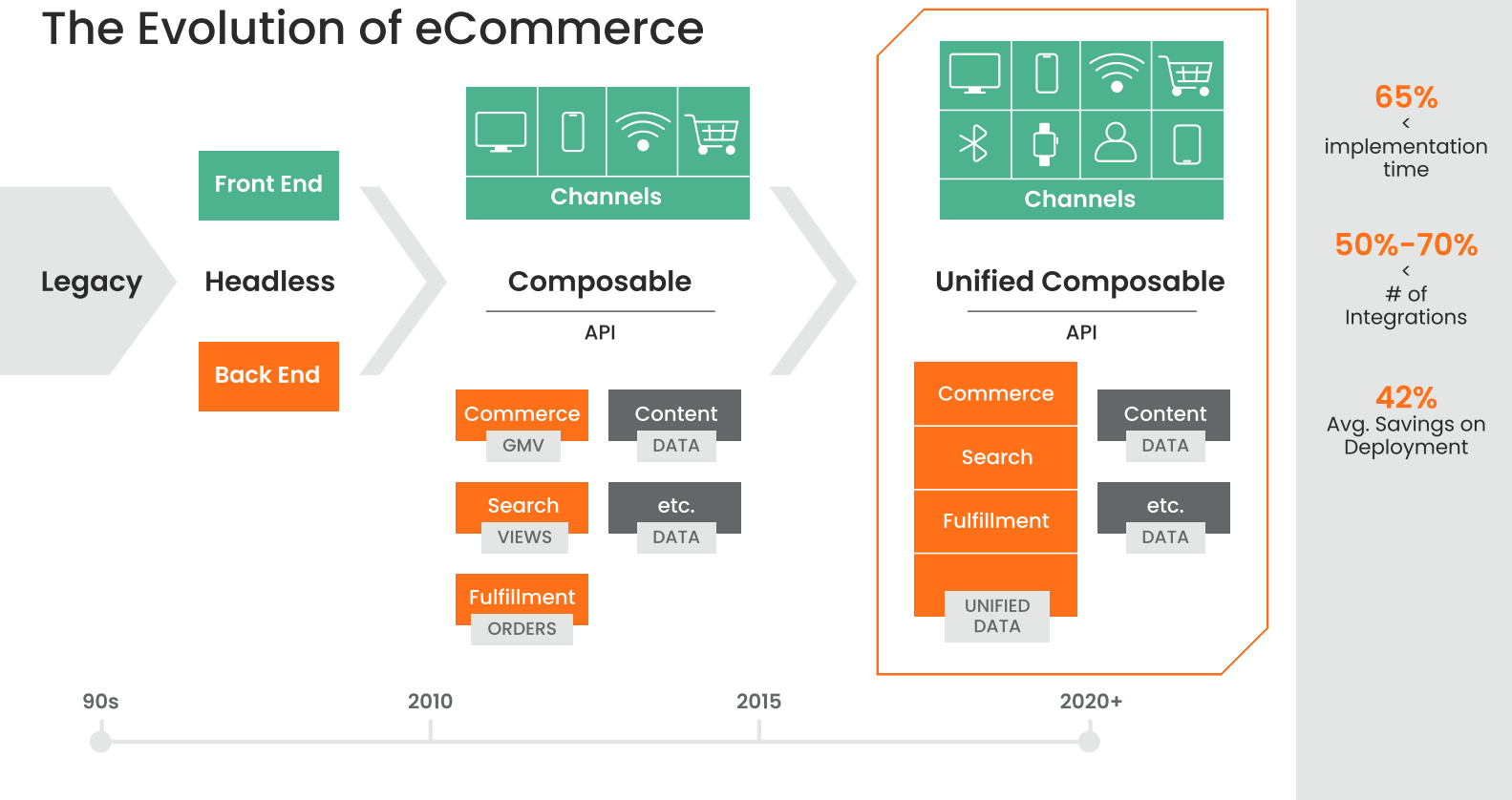


Fig 3. Illustrated here is the transformative nature of the evolution of headless and composable commerce. The headless era focused on separating the front-end from the back-end, allowing for greater flexibility. This was followed by the composable era, which broke down the back-end into individual, interchangeable components. Now, with solutions like KIBO, these advancements are unified. The platform offers a complete suite of composable components along with a robust front-end, enabling businesses to create a seamless, true multichannel experience by bringing all elements together.

Why is it Gaining Traction?

The rise of composable commerce is a direct response to the limitations of traditional, monolithic platforms. In today's dynamic digital landscape, businesses need to:

- **Innovate Faster:** Composable commerce allows for a rapid time-to-market. When a new feature or a new channel is needed, you can develop and deploy a single component without a full-scale replatforming project.
- **Offer Unique Experiences:** With the "best-of-breed" approach, you can create a highly tailored and personalized customer experience by integrating the most advanced solutions for search, personalization, and content.
- **Remain Agile:** The ability to swap out components means you are not locked into a single vendor's roadmap. If a new technology emerges that is better than what you are using, you can adopt it with relative ease, future-proofing your business.

In essence, composable commerce is not just a technology trend; it's a strategic shift that puts the business in control of its digital future, empowering it to build a highly customized, scalable, and adaptable platform that can evolve at the pace of modern commerce.



The way people are buying is changing... and that requires an architecture that can be pieced together, not one you have to slap together from one vendor and pay out the nose for. That's a thing in the past now.

Zach Zalowitz, Principal Order Management Practice



Why Composable Platforms are Ideal for Migration

When it comes to the complex task of migration, composable platforms offer a strategic advantage over traditional monolithic systems. Their flexible, modular design makes the journey less risky and more efficient, ultimately setting you up for long-term success.

A composable platform is a superior choice for a commerce migration for several key reasons, primarily centered on its flexibility, agility, and long-term viability. When compared to a traditional monolithic platform, which is a single, tightly integrated system, a composable approach offers a more strategic and less disruptive path to a modern commerce solution.

Here's a breakdown of why a composable platform is more advantageous for a commerce migration:

1 Phased and Less Risky Migration

A monolithic system often requires a "big bang" migration, where the entire platform is replaced at once. This is a high-risk strategy because any errors or issues affect the entire site, leading to potential downtime and lost revenue.

In contrast, a composable platform, with its modular architecture, allows for a phased migration. You can migrate specific functions or "microservices" one at a time, such as your pricing, promotions, cart and checkout, real-time inventory visibility, Product Information Management (PIM) system or your Content Management System (CMS). This approach, sometimes called the "strangler pattern," allows you to:

- Minimize risk: You can test and validate each new component in a live environment before fully committing to the migration. If a component doesn't work as expected, you can roll it back without impacting the rest of your business.
- Reduce disruption: You can run your old and new systems in parallel, gradually shifting traffic and functionality to the new platform, ensuring a smooth transition for your customers.

2 "Best-of-Breed" Solutions

Monolithic platforms often force you to use the features and integrations provided by a single vendor. While this may be convenient at first, it can limit your ability to innovate and stay competitive. Composable commerce is built on a "best-of-breed" approach. This means you can select the best possible solution for each specific business need from different vendors. For example, you can choose:

- A powerful CMS for content management.
- A specialized search engine for product discovery.
- An AI-driven personalization engine for recommendations.
- A robust payment gateway.

This allows you to create a technology stack that is perfectly tailored to your unique business requirements, rather than settling for a "good enough" solution.

3 Agility and Faster Time-to-Market

In a monolithic system, making a change to one part of the platform can have unintended consequences on other parts, leading to lengthy and complex development cycles. A composable platform's decoupled nature allows for unparalleled agility. Each component is independent, communicating with others through APIs. This means:

- **Faster innovation:** You can quickly and independently develop and deploy new features, such as a new checkout flow or a mobile app, without having to rebuild the entire system.

- **Rapid response to market changes:** You can easily adopt new technologies or swap out old components for better ones to keep up with evolving customer expectations and market trends.
- **Empowered teams:** Different development teams can work on separate components simultaneously, streamlining workflows and accelerating the pace of innovation.

4 Scalability and Cost Efficiency

Scaling a monolithic platform often means scaling the entire system, even parts that aren't under heavy load, which can be inefficient and costly. With a composable architecture, you can scale individual components as needed. For example, during a flash sale or peak season, you can allocate more resources to your search or checkout components to handle the surge in traffic, while keeping other parts of your system at their normal capacity. This "pay-for-what-you-use" model helps optimize costs and improve performance.

5 Future-Proofing and Reduced Vendor Lock-in

A monolithic platform often leads to "vendor lock-in," where you are tied to a single vendor's roadmap and technology. If the platform doesn't evolve as you need it to, you are faced with another costly and disruptive replatforming project.

A composable platform gives you the freedom to choose and switch vendors for specific components. This reduces your dependence on any one vendor and allows you to continuously evolve your technology stack to meet future needs, making it a truly future-proof solution.

Leveraging Integrated Modularity for Strangler Pattern Migrations

Integrated modularity directly supports the strangler migration pattern by providing a practical, low-risk way to replace a legacy system's functionality piece by piece. Instead of a costly, high-risk "big bang" overhaul, this approach allows you to strategically add modern, modular components to your existing platform.

How it Works

As mentioned earlier in the guide, the strangler pattern is a phased migration technique where you incrementally "strangle" an old, monolithic system by redirecting traffic to new, standalone microservices. Integrated modularity provides the perfect toolset for this.

This approach works by building a new service for a specific function and then redirecting traffic for that function to the new system. The old system remains in place for all other functions. As you build and deploy more new services, you redirect more and more traffic, slowly replacing the old monolith's capabilities until it can be safely retired.

For example, a platform like KIBO can co-exist with your existing system while phasing in new functionality using robust APIs. You could implement a new [order management system \(OMS\)](#) in a phased approach, first adding in a microservice for [inventory visibility](#), then a separate one for [order routing](#), and so on. This allows you to:

- **Replace Functionality:** The new module handles a specific function, allowing you to gradually turn off or "strangle" the corresponding feature in your old system.

- **Reduce Risk:** You can test each new module in a live environment without impacting the rest of your business. If a new module doesn't perform as expected, you can revert to the legacy system with minimal disruption.
- **Enable Incremental Investment:** You can spread the cost and effort of migration over time by investing in one module at a time, continuously adding new capabilities and moving toward a fully modernized architecture at your own pace.

Essentially, integrated modularity makes the strangler pattern a reality. It offers a practical middle ground between a rigid monolith and a complex, all-in-one composable stack. You get the stability of your legacy system while gaining the agility and innovation of modern, modular components.

“

We need to agree on on patterns and protocols so that software in our space that is composable is also interoperable with other solutions because only then do brands have the freedom to really mix and match and swap out if and when they need.

*Sonja Keerl, Co-founder
of the MACH Alliance*

MACH
ALLIANCE



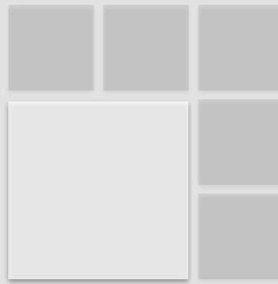
Composability is a spectrum

There is no “one-size fits all”



All-In-One Solution

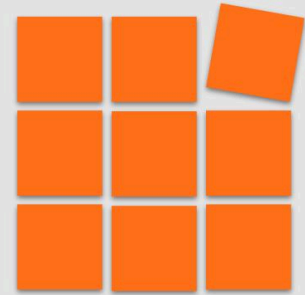
Monolith, unintegrated



Integrated Composable

Monolith, with several composable modules

KIBO UNIFIED COMMERCE



Fully Composable

Completely modular with fully replaceable systems



Fig 4. Composability exists on a spectrum, offering flexibility beyond a complete platform migration. Many companies can benefit from a middle-ground approach by integrating composable point solutions with their existing systems. KIBO's platform is so modular, it enables customers to strategically acquire and deploy specific components to augment and extend the life of their current technology stack. This allows for targeted innovation without the disruption of a full-scale overhaul.

KIBO for Unified Commerce

The [KIBO platform](#) is fully composable, giving businesses the flexibility to adopt a unified commerce strategy by leveraging all of KIBO's components. This means that while the platform remains modular under the hood, all of its capabilities run on the same underlying tech stack, providing a single, cohesive, and seamless solution.

This shared foundation is a significant advantage. Because KIBO's entire platform is built on the same architecture, there are no acquisitions to integrate and no complex integrations required between products. Businesses can select the functionality they need most, confident that it will work seamlessly with their existing stack.

This approach offers distinct benefits for Integrated Modularity:

- **Flexible Deployment:** Pick and choose only the functionality you need.
- **Innovation Without Overhaul:** Extend your platform without a risky, expensive replatforming.
- **Incremental Migration:** Modernize your stack with a painless, step-by-step approach.

A Closer Look at the KIBO Platform

Composable Microservices You Can Deploy Incrementally

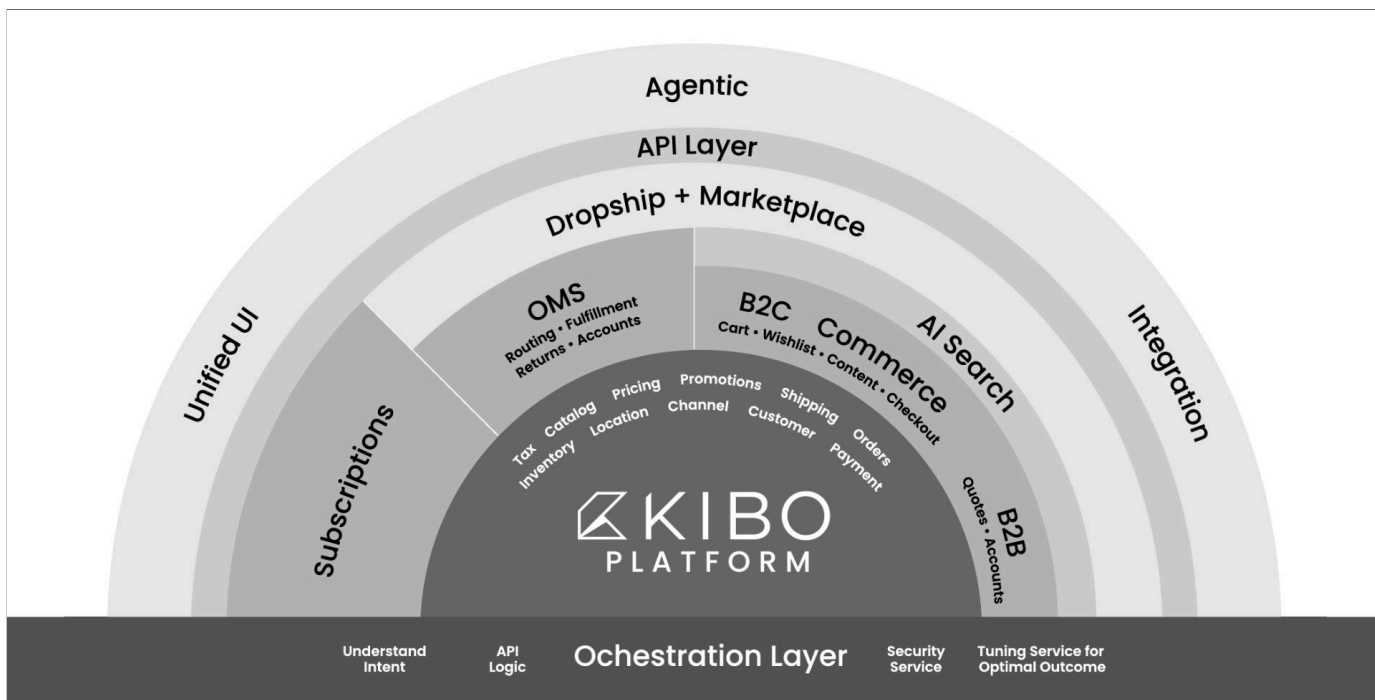


Fig 5. KIBO's products are built on a modular foundation of microservices, allowing new functionality to be unlocked by simply extending the existing framework. Each product uses a shared set of objects and microservices, so no complex integrations or connectors are needed. This unified approach gives customers a broad range of solutions to grow their commerce operations seamlessly on a single platform.

The KIBO platform's true power lies in its **modular and shared-service nature**. At its core are shared objects and microservices that form a foundational layer across all products. As customers build their infrastructure, they can seamlessly extend their capabilities. For instance, if you have your product catalog and pricing established for ecommerce, adding a new function like real-time inventory management or order routing is as simple as layering new microservices onto the existing ones. This shared architecture significantly accelerates implementation time for new products.

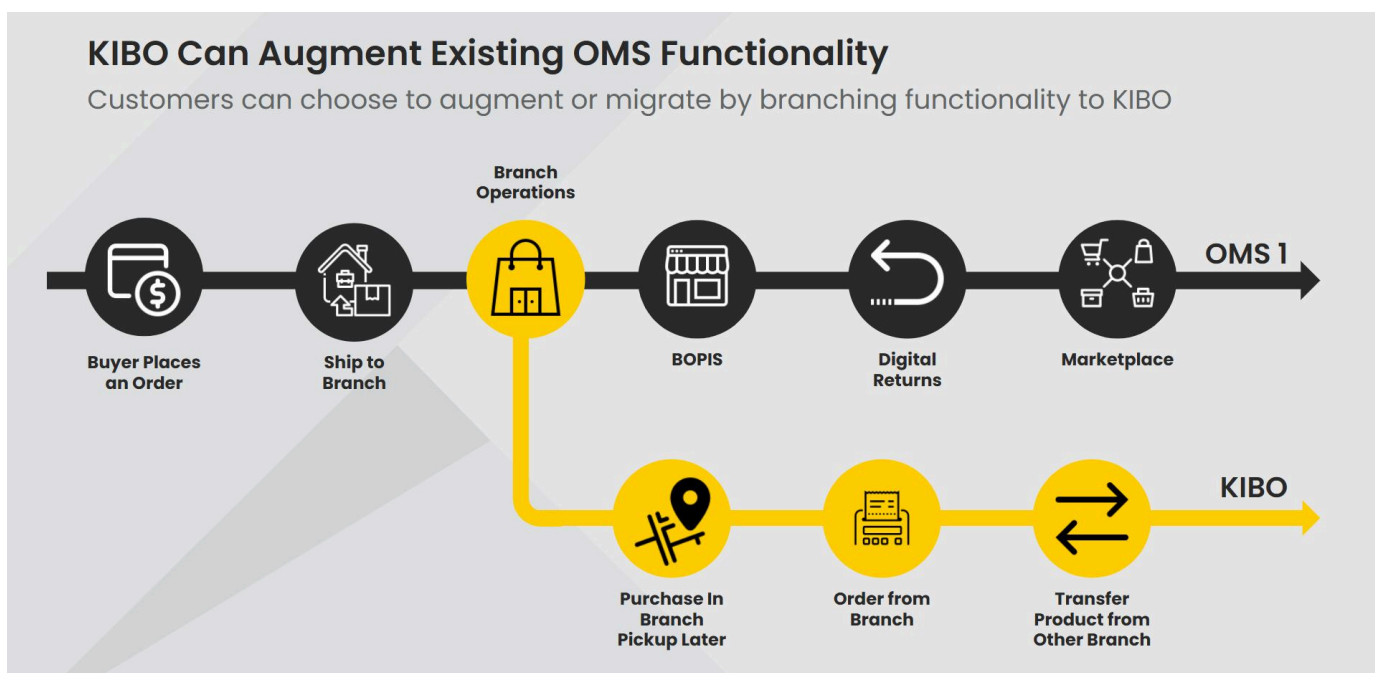


Fig 6. The KIBO Platform enables a phased migration by allowing you to incrementally shift functionality away from your incumbent system. KIBO can start by taking on additive new features or by replacing existing modules as needed. This approach allows you to augment your current OMS without a high-risk replatforming project.

We've also developed specific solutions that build on this foundation. Our [Dropship](#) and [Marketplace](#) capabilities, for example, are extensions of our core commerce platform. We've created an EDI layer to integrate third-party logistics (3PL) into traditional OMS functions, enabling real-time inventory and complex order routing. For marketplaces, we provide vendor and supplier portals to manage your third-party vendors as an extra commerce experience.

All of this functionality is unified by a robust API architecture that allows you to "plug and play" KIBO into any existing environment.



We are rethinking how modern commerce gets done. Our roadmap is laser-focused on solving real operational pain points for our clients. Whether it's AI agents that automate merchandising or reverse logistics workflows that cut costs, our goal is to deliver smarter tools that scale with our customers' businesses.

Sachin Sharma, Chief Product Officer



Why Composable Platforms are Ideal for Migration

The future of commerce is intelligent and autonomous. As you plan your migration, you need to consider how your new platform will support the next generation of customer experiences, powered by AI-driven "agentic commerce."

Solutions like [KIBO Agentic Commerce](#) move beyond simple automation to create intelligent, autonomous agents that can act on a user's behalf. This includes everything from a conversational AI that can complete a purchase from start to finish to an intelligent reordering system that anticipates a customer's needs. This technology is not a bolt-on feature; it fundamentally changes the customer experience by:

- **Personalizing the Experience:** Agents can learn from customer behavior and preferences to offer hyper-relevant product recommendations and services.
- **Automating Tasks:** They can handle routine tasks like order tracking, returns, and reordering without human intervention, freeing up customer service teams.
- **Driving Proactive Engagement:** Agents can initiate conversations with customers, provide timely updates, and even anticipate a need before it's expressed.

Migration Considerations

When migrating to a new platform, it's crucial to assess how it will enable or hinder your move towards agentic commerce.

1. API-First and Headless Architecture

Agentic commerce relies on a seamless connection between the AI layer and the core commerce system. A headless and API-first architecture is essential. The AI agent (the "head") needs to be able to access all the business logic and data (the "body")—such as product information, inventory levels, and customer profiles—through robust APIs. A traditional, monolithic platform with tightly coupled front-end and back-end will make it difficult, if not impossible, to integrate these intelligent agents effectively.

2. Data and Analytics

Agents are only as smart as the data they are trained on. A successful migration should include a strategy for collecting, cleaning, and structuring data to feed these AI models. Consider:

- a. Data Consolidation:** Will the new platform allow you to unify data from all touchpoints into a single source of truth?
- b. Real-time Data Access:** Can the AI agents access real-time data to make instant decisions on inventory, pricing, or product availability?
- c. Data Security and Privacy:** How will the new platform handle sensitive customer data to ensure it complies with privacy regulations?

3. Composable vs. Monolithic Platforms

When migrating, one of the most significant choices you'll make is selecting an architecture—a decision

that will determine how easily you can adopt future technologies like agentic commerce.

a. Composable Platforms: A

composable architecture is an ideal fit for agentic commerce. Its modular, best-of-breed nature allows you to plug in a specialized AI engine or an automation tool as a microservice. You can experiment with different agents and easily swap them out as the technology evolves without needing a full replatform.

- b. Monolithic Platforms:** These systems are often rigid and limit your ability to integrate cutting-edge AI. You may be forced to use the platform's native (and often limited) AI capabilities or undertake a complex and costly custom integration.

4. Total Cost of Ownership (TCO)

While a composable, agentic-ready platform may have a higher initial setup cost, it can lead to significant long-term savings. The ability to automate tasks and optimize experiences with AI agents can:

- a. Reduce operational costs:** Less need for human intervention in customer service and manual tasks.
- b. Increase revenue:** Enhanced personalization leads to higher conversion rates and customer loyalty.
- c. Improve efficiency:** Faster, more accurate transactions powered by AI.

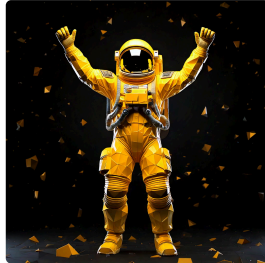
In summary, the consideration of agentic commerce should be a core component of any modern commerce migration plan. It's a key factor in future-proofing your business and ensuring your new platform can support the next generation of customer experiences.

KIBO's Agentic Layer: The Next Leap in Commerce

Further enhancing the composable experience is KIBO's Agentic Layer. This multi-agent approach provides modular AI agents for every part of the KIBO platform to help you accelerate innovation. The agentic layer is accessible to all objects and microservices, with existing and planned rollouts for a variety of functions, including:



Shopper



Customer Success



Merchandising



Pricing/Promotions



Order Routing



Forecasting



Fulfillment



Reverse Logistics

These agents operate on a spectrum of capabilities, from the ability to engage with users and explain data, to boosting productivity and, most critically, leveraging AI to tune and optimize your system. For example, if you have specific safety stock metrics set by the store, an optimization agent can automatically adjust those parameters against your business goals.

A crucial underpinning of these agents is our powerful search engine. Many retailers make the mistake of deploying an agentic solution that relies on a simple synonym or keyword-based search. This approach is limited. [KIBO's vector AI search](#) dramatically increases the accuracy of agent responses by incorporating all product attributes in your catalog. This ensures the agent not only understands what you're asking for but can also deliver precise and accurate results.

In Summary

For a successful ecommerce migration, you need careful pre-planning and a strategic approach. Companies must choose between a high-risk, all-at-once Big Bang strategy and a lower-risk, gradual Phased migration.

Modern commerce is moving towards a composable architecture built on MACH principles. This provides the agility, scalability, and flexibility needed to support those phased approaches. Looking ahead, this modular foundation is also key for adding agentic commerce. This means using AI-powered agents to automate tasks and personalize the customer experience. Ultimately, this helps future-proof your platform and drives long-term efficiency and growth.

About KIBO

KIBO is a composable commerce platform for retailers, manufacturers, distributors, and wholesalers who want to simplify complexity and deliver modern customer experiences. KIBO is the only modular commerce platform supporting unified experiences across Order Management, eCommerce, and Subscriptions. Companies like Zwilling, Ace Hardware, Boscov's, Nivel, and REEDS Jewelers trust KIBO to bring sophistication and simplicity to their commerce operations. Learn more at <https://kibocommerce.com>.

About the Authors:



Natalija Pavić

Senior Director of Product Marketing at KIBO

Natalija Pavić is the Product Marketing Leader at KIBO Commerce where her team handles product market messaging including content, social, public relations, and analyst relations. She is an ecommerce expert and a thought leader on the topic of the future of ecommerce and has been featured on numerous podcasts including Martalks, OmniTalk, Ecommerce Coffee Break, Retail Checks and Balances, Digital Shelf Institute, and AI with Sacha. She is also an AI expert and inventor with a patent on generative promotions and is patent pending on two more AI innovations.



Shannon Abel

Sr. Corporate Marketing Manager

For over seven years, Shannon has worked in the commerce technology industry—first with Blue Acorn iCi, then joined KIBO in 2022. As the corporate marketing manager, she manages KIBO's content, PR, and brand strategies. Shannon graduated from Clemson University in 2014 and enjoys spending her free time with her husband, two dogs, and horse in Charleston, SC.