WHITE PAPER · 2026

REDIFY

ECOMMERCE ARCHITECTURE · B2B MANUFACTURING.

Headless, Composable or MACH?

Decision Framework for Enterprise Ecommerce Architecture in Manufacturing



TABLE OF CONTENTS

Document Structure

1. Executive Summary 03

In B2B manufacturing, choosing the right architecture is a strategic decision.

2. What Does "Commerce Architecture" Mean in an Enterprise Manufacturing Context? 04

How digital technologies support sales, order management, and integration with ERP and CRM.

3. Architecture Map: Monolithic → Headless → Composable → MACH 05

The evolutionary stages of ecommerce architecture explained clearly.

4. Headless vs Composable: What's the Difference? 07

Frontend flexibility vs full stack modularity.

5. Decision Tree: Which Architecture for Which Business Scenario? 08

A framework for choosing based on business needs and tech debt.

6. Composable vs Monolithic: When Does Migration Make Sense? 09

The signals that indicate it's time to change.

7. Ecommerce Replatforming: How to Protect SEO, GEO and Organic Visibility 09

Migration should not cost you organic visibility.

8. Comparison of the Major Platforms 11

Comparative evaluation for CTOs and CIOs.

9. How to Map Your Current Tech Debt Before Deciding 12

Metrics and indicators for a concrete mapping.

10. Redify Assessment: How to Evaluate an Architecture in 30 Minutes 13

A quick method to understand where you are and where you want to go.

11. Checklist: Questions to Ask Before Choosing Your Ecommerce Architecture 14

The right questions to make the right choice.

12. Conclusions 15

Headless, composable and MACH are tools, not goals.

WHITE PAPER · 2026

1. Executive Summary

In B2B manufacturing, "commerce architecture" does not just mean choosing an ecommerce platform, but defining how the front-end, digital ordering, ERP, CRM, product configurators and production systems work together. The transition from monolithic to headless commerce, composable commerce or MACH architecture is a strategic choice that impacts resilience, time-to-market and tech debt management.

> **This whitepaper provides a decision framework for CTOs and CIOs, with the aim of explaining the difference between headless and composable commerce architecture, clarifying when it makes sense to migrate from a monolithic to a composable platform, objectively comparing the major platforms (HCL Commerce, Salesforce B2C, Adobe Commerce, commercetools, VTEX), and proposing a decision tree to choose the right ecommerce architecture for a manufacturing scenario. The Redify Assessment shows how to evaluate the current architecture and tech debt status in 30 minutes, while the final checklist offers the key questions to ask before committing budget and roadmap.**

WHITE PAPER · 2026

2. What Does "Commerce Architecture" Mean in an Enterprise Manufacturing Context

In an enterprise context, "commerce architecture" is the set of choices that define how digital supports orders, product configuration, pricing and relationships with dealers and branches. It is not just an IT topic, but an operating model decision.

In B2B manufacturing, commerce architecture covers:
• Digital channels (dealer portals, branch portals, B2B/B2C sites, apps, OEM APIs).
• Core systems (ERP, CRM, PLM, configurators, pricing systems) that must reliably communicate with the front-end.
• Data flows (catalog, prices, orders, availability, customer records) between systems.
• Business rules (commercial terms, compatibility rules, distribution logic) that must be consistent across all touchpoints.

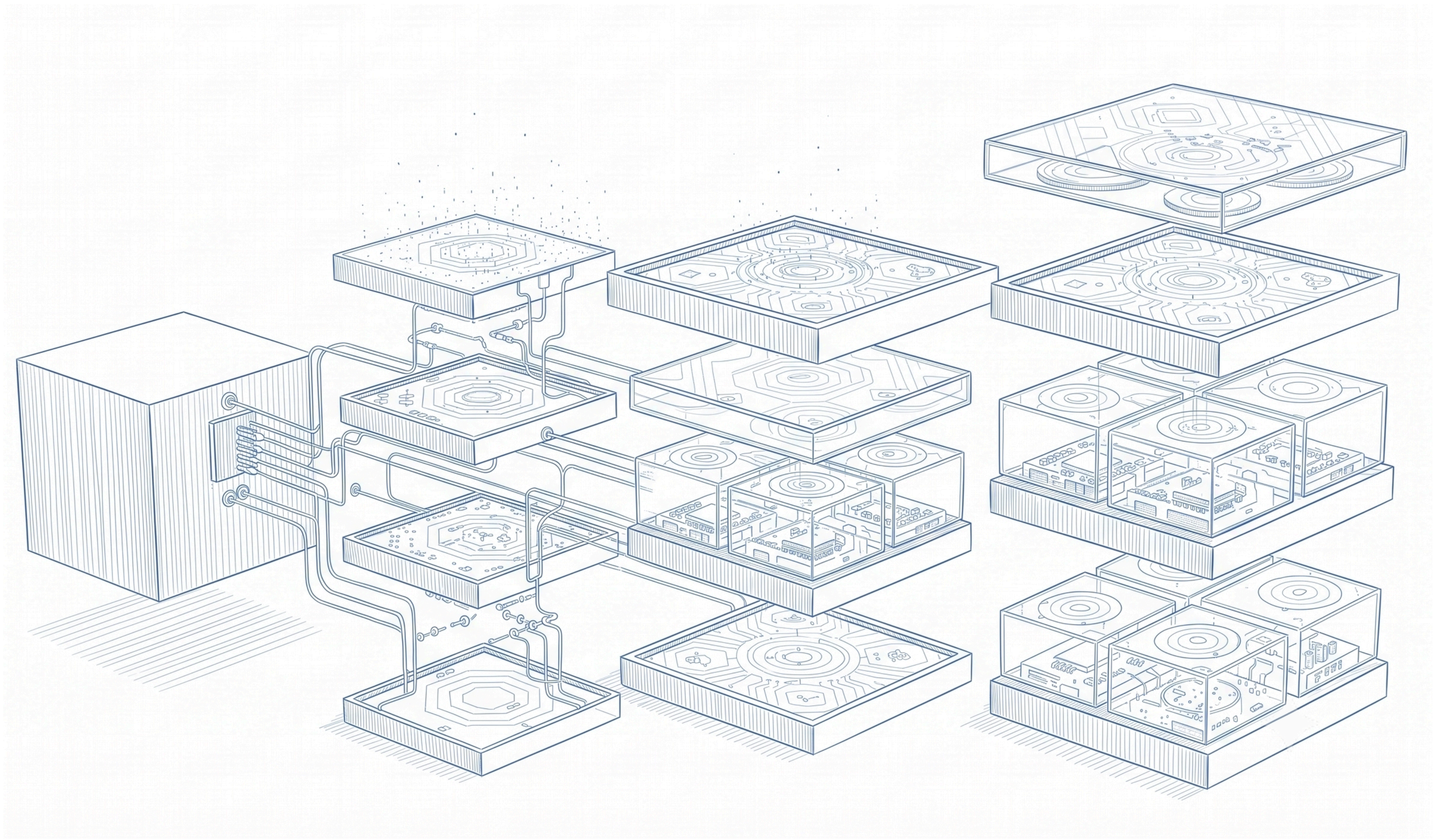
When a Redify team works with a CTO, the starting point is not "which platform", but "how we want the architecture to support the business". The headless, composable or MACH framework is chosen based on these goals.

WHITE PAPER · 2026

3. Architecture Map: Monolithic → Headless → Composable → MACH

Monolithic, headless, composable and MACH are not marketing labels, but evolutionary stages of ecommerce architecture. Understanding where your stack sits today is essential to deciding the next step.

Monolithic	Headless	Composable	MACH
Single codebase, tightly coupled components	Separate frontend, API-based communication	Modular stack, best-of-breed services	Microservices, API-first, cloud-native, headless



Below you will find a more in-depth analysis of each architectural stage, to understand where your stack sits today and what the next step might be.

WHITE PAPER · 2026

Monolithic

A monolithic ecommerce architecture is characterized by:

• Front-end and back-end tightly coupled in a single application.

• Catalog, cart, checkout, promotions and CMS modules integrated in the core.

• Changing one part often means touching everything, impacting stability and release times.

In manufacturing, many legacy ecommerce platforms fall into this category: ERP-centric portals, custom solutions that manage everything in a single codebase.

Headless commerce

Headless commerce separates the front-end ("head") from the commerce back-end:

• Front-end (site, app, dealer portal) developed independently.

• Back-end that exposes APIs for products, prices, cart, orders.

• Advantages: greater UX flexibility, multi-channel support, ability to innovate the front-end without rewriting the back-end.

For a manufacturing company, headless ecommerce makes sense when the back-end is relatively solid (stable ERP and commerce engine) but you want to accelerate on UX, dealer portals, apps and web configurators.

Composable commerce

Composable commerce goes beyond headless: it does not just decouple the front-end from the back-end, it makes the entire stack modular:

• Every capability (search, catalog, pricing, promotions, checkout, OMS, CMS) is a separate component.

• "Best-of-breed" services from different vendors (e.g. specialized search, pricing engine, order engine) are combined via API.

• Advantages: maximum flexibility, resilience (one component failing does not block everything), ability to evolve piece by piece.

In manufacturing, composable commerce allows you to integrate configurators, advanced pricing systems and dealer channels in a modular way, without being locked into a single vendor.

WHITE PAPER · 2026

MACH architecture

MACH architecture (Microservices, API-first, Cloud-native, Headless) is the technical paradigm that often underlies composable solutions:

Microservices

Functionalities broken down into autonomous services.

API-first

Everything exposed via API, integrable with other systems.

Cloud-native

Designed to scale horizontally and leverage distributed cloud.

Headless

Front-end separated from business services.

Not all composable solutions are fully MACH, but many modern platforms (e.g. commercetools) adopt these principles to offer flexibility and scalability.

4. Headless vs Composable: What's the Difference?

Headless commerce primarily solves the problem of front-end flexibility; composable commerce addresses the entire architecture, making the back-end modular as well. The choice depends on the type of constraints the CTO is looking to overcome.

Headless commerce

Focus: decoupling frontend and backend.
Main benefit: freedom on the presentation layer (sites, apps, portals) while maintaining an existing commerce engine.
When it makes sense: when the main problem is UX, time-to-market of the front-end, support for new channels, while the backend (ERP + order engine) is stable.

Composable commerce

Focus: modularizing the entire stack, not just the front-end.
Main benefit: ability to choose and replace individual services (search, pricing, OMS, CMS) with best-of-breed, reducing vendor lock-in.
When it makes sense: when you also need to rethink the backend, order flows, catalog management and pricing, integrating multiple specialized systems.

> **For an enterprise retailer or a manufacturer with an established B2B channel, the "headless vs composable" question translates to: "Do I just need to free up the front-end, or do I want to make the entire business engine modular?"**

WHITE PAPER · 2026

5. Decision Tree: Which Architecture for Which Business Scenario?

There is no "right architecture" in absolute terms; some architectures are better suited to specific business and tech debt scenarios. A decision tree helps a CTO position their own context.

Nodo 1

Where the Problem Is Concentrated

Key questions:

• Is the main constraint in the UX and front-end (portals, sites, apps), or in the backend (ERP/CRM integrations, performance, order flows)?

• Is the current catalog, pricing and order engine stable and well-integrated, or is it a source of errors and slowdowns?

If the problem is almost exclusively on the UX side, moving to headless ecommerce may be sufficient.

If problems are spread across the entire stack, it is more coherent to explore composable commerce.

Nodo 2

Organizational Maturity and Competencies

Questions:

• Is the internal team ready to manage an ecosystem of multiple vendors and services, or does it prefer a single stack?

• Is there experience with APIs, microservices, observability, DevOps?

Composable commerce requires greater maturity in multi-vendor integrations and governance; in less mature contexts, moving to headless on a robust platform can be a first step.

Nodo 3

Time-to-Market vs Depth of Transformation

Questions:

• How urgent is it to improve UX and the digital channel (e.g. dealer pressure, competitors)?

• How much budget and time is available for an in-depth transformation of the ecommerce architecture (12-24 months)?

If the urgency is high and the transformation window is limited, it makes sense to adopt an incremental strategy: headless first, composable in subsequent steps.

WHITE PAPER · 2026

6. Composable vs Monolithic: When Does Migration from a Monolithic Platform Make Sense?

Migrating from a monolithic to a composable platform makes sense when the cost of tech debt and architectural constraints exceeds the cost of orchestrating more services. The decision must start from an objective mapping of technical debt.

Technical debt manifests as:

- Technical debt manifests as:
- • Modules that cannot be modified without breaking other parts.
- • Poorly documented or fragile integrations.
- • Legacy libraries and frameworks that are difficult to update.
- • Manual workarounds and hidden operational scripts.

Signs that "monolithic" is no longer sustainable:

- Signs that "monolithic" is no longer sustainable:
- • Every release is perceived as risky.
- • The team avoids touching certain parts of the system.
- • Time spent on bugfixes exceeds time spent on new features.

In these cases, composable commerce (eventually supported by MACH) allows you to isolate the problematic parts and progressively replace them with modular services.

7. Ecommerce Replatforming: How to Protect SEO, GEO and Organic Visibility During Migration

In a replatforming project, the loss of organic visibility is not an unavoidable side effect, but an architectural and operational risk that must be managed from the start. When URLs, templates, content, structured data and rendering logic change without SEO governance, the cost is not measured only in lost rankings, but in qualified traffic, leads and commercial continuity.

In an enterprise context, platform migration is not just about moving from one commerce engine to another. It involves changes to URL structure, how pages are rendered, metadata management, internal linking, category and product templates, and often the logic by which content and data are orchestrated between CMS, PIM, ecommerce engine and front-end. For this reason, SEO and GEO should not be treated as a final quality check, but as a project layer that accompanies discovery, design, build and cutover.

The five minimum coverage areas are:

Complete mapping of legacy and new URLs, with consistent and verified 301 redirects.

Validation of title, meta description, canonical, hreflang, heading hierarchy and structured data before go-live.

Verification of actual page rendering, especially in headless or composable architectures where JavaScript, APIs and front-end can alter indexing.

Protection of high-value organic pages, such as categories, strategic product pages, brand pages and editorial content.

Post-launch monitoring with Google Search Console, analytics and control of crawling, ranking and traffic anomalies in the first weeks.

In fashion, retail and luxury sectors, this risk is even more acute, because organic visibility is often spread across seasonal categories, collections, brand landings and high commercial-pressure pages. In these cases, a technically correct but SEO-fragile replatforming can undermine months of work on assortment, content and organic acquisition.

For this reason, when the Redify team evaluates a migration, the focus is not only on "which architecture to choose", but also on "how to guarantee continuity of visibility during the transformation".



- Replatforming can cause loss of organic visibility if SEO and GEO are involved too late in the project.
- URLs, rendering, metadata, schema and internal linking must be verified before go-live.
- In fashion, retail and luxury, the risk is higher due to seasonal categories, collections and high-value commercial content.

WHITE PAPER · 2026

8. Comparison of the Major Platforms

Platforms differ in architectural approach, target (B2B/B2C), composability level and ecosystem.
A CTO must evaluate them not just for features, but for alignment with their own architecture strategy.

HCL Commerce

- Monolithic origins, with evolutions towards headless support and API integrations.
- Strong presence in enterprise contexts, often integrated with complex ERPs.
- In manufacturing, it is often part of the legacy ecommerce platform, with modernization projects towards more modular models.

Salesforce B2C Commerce

- Cloud platform for B2C retailers and brands, with a strong marketing and CRM ecosystem.
- Approach that supports headless, with APIs for integration with custom front-ends and external services.
- Suited to scenarios where CRM and marketing are central and you want to leverage the Salesforce suite.

Adobe Commerce (ex Magento)

- Strong B2B vocation and customization flexibility; possibility of adopting headless patterns.
- Wide ecosystem with extensions and integrations, but watch out for tech debt from unmanaged customizations.

Commercetools

- Natively MACH and composable platform, oriented to microservices, API-first and cloud-native.
- Offers modular building blocks for cart, checkout, promotions, order management, integrable with external systems.
- Suited to organizations ready to manage fully composable and multi-vendor architectures.

VTEX

- Cloud platform with ecommerce and marketplace functionality, with headless and API support.
- Oriented to retail and B2B scenarios where you want to combine store and marketplace.

> **When a Redify team works with a CTO, the comparison is not reduced to "which is best", but to "what is the ecommerce architecture consistent with tech debt, objectives and internal capabilities".**

WHITE PAPER · 2026

9. How to Map Your Current Tech Debt Before Deciding

Without a tech debt mapping, any discussion on headless, composable or MACH risks being abstract. You need metrics and indicators to assess how much the current architecture is limiting the business.

Techniques for Mapping Technical Debt

- Critical module inventory: identify components managing catalog, pricing, orders, ERP/CRM integrations.
- Fragility analysis: identify parts of the system that cause frequent regressions when touched.
- Integration mapping: document all flows between ecommerce, ERP, CRM, OMS, configurators, paying attention to non-standard integrations.
- Collection of operational evidence: incidents, resolution times, maintenance effort.

Critical Debt Indicators

- "Untouchable" modules that require weeks for small changes.
- ERP or CRM integrations based on undocumented scripts.
- Parts of the code that no one on the team wants to maintain.

When a Redify team works with a CTO, the result of this analysis is a map showing where to focus the transformation: headless on the front-end, composable on the most critical modules, or gradual modernization via MACH patterns.

WHITE PAPER · 2026

10. The Redify Assessment: How to Evaluate an Architecture in 30 Minutes

A brief assessment does not replace a project, but it helps understand whether it is worth going deeper. In 30 minutes, a CTO can get an initial reading of the risks and opportunities of their own ecommerce architecture.

Context

Current platform (monolithic, headless, hybrid), scope (B2B, B2C, dealer, countries).

Stress signals

Incidents, time-to-market, critical integrations, perceived technical debt.

Business objectives

Digital growth, new channels, improved dealer experience.

Possible architectural trajectories

Progressive modernization via MACH patterns.

Following this conversation, the CTO receives a qualitative summary of the main trade-offs and areas to explore further.

WHITE PAPER · 2026

11. Checklist: Questions to Ask Before Choosing Your Ecommerce Architecture

The final checklist allows CTOs and CIOs to translate the discussion on headless commerce, composable commerce and MACH architecture into concrete questions. The goal is to avoid hype-driven choices.

Questions on context and objectives

- Which markets, customer segments and digital channels do we want to serve in the next 3–5 years?
- What is the role of the digital channel in the business model (dealer orders, self-service, configuration)?
- What are the main pain points today (UX, performance, order errors, integrations)?

Questions on current architecture

- Is the current platform primarily monolithic, headless or hybrid?
- Which components are most fragile or costly to maintain (catalog, pricing, orders, integrations)?
- Are there already clean, documented APIs for integration with external systems?

Questions on skills and organization

- Does the internal team have experience with APIs, microservices and multi-service orchestration?
- Are DevOps processes, observability and monitoring tools in place?
- Is there capacity to manage a composable commerce setup with multiple vendors, or is a single stack preferable?

Questions on risks and roadmap

- What is the board's risk tolerance for architectural changes?
- Which milestones are realistic at 6, 12 and 18 months?
- Has a rollback plan and SEO mitigation strategy for the migration been considered?

12. Conclusions

Headless commerce, composable commerce and MACH architecture are tools, not objectives. The decision-making framework proposed in this whitepaper allows CTOs and CIOs in manufacturing to choose the ecommerce architecture based on objectives, tech debt and internal capabilities, rather than following market labels.

> **A digital architecture partner like Redify helps transform this framework into an operational and strategic path: rapid assessment, technical debt mapping, definition of a realistic roadmap (0–18 months) and building a business case understandable for CFO and board.**

WHITE PAPER · 2026

Have you recognized any of these signals in your ecommerce architecture?

If during this whitepaper you recognized signals that resemble your ecommerce architecture situation in manufacturing, the next step is not to immediately choose between headless, composable or MACH, but to clarify objectives and constraints.

If you want to understand before deciding, start a conversation by booking your free 30-minute Discovery Call with the Redify team. We will discuss the current state of your architecture, the risks and the possibilities of evolving towards headless commerce, composable commerce or MACH architecture.

redify.co/discovery

REDIFY

FURTHER READING

redify.co/insights

Luxury Commerce

EDITION

v1.0 · 2026

English edition

© REDIFY 2026