OSF | DIGITAL

Navigating Technology Investments: The Goldilocks Approach to Headless and Composable

osf.digital

Executive Summary

As businesses plan for their next peak, many face critical decisions on technology investments that balance innovation with measurable outcomes. Two terms in the spotlight right now—headless and composable—are often promoted as essential upgrades, yet they are frequently misunderstood, leading to costly mistakes.

Common pitfalls include overinvesting in complex systems that don’t directly support conversion rate improvements or agility, overestimating benefits of headless setups alone for speed, and undertaking high-maintenance integrations that lack a clear ROI.

A balanced, “Goldilocks” approach—focused on direct business outcomes rather than technical zeal—can avoid these risks.

Companies should prioritize technology decisions that demonstrably drive revenue growth, manage costs, and support agility aligned with business goals. Starting with clear, business-driven objectives ensures a sustainable and valuable architecture choice, avoiding costly technical detours, and we will show you how.

Table of Contents

Understanding Key Terms: Composable and Headless 4

Positives Messages in the Market 5

Negative Messages in the Market 7

Adopting a Balanced Approach: The “Goldilocks” Strategy 10

- Start with the business 10
- PWA Kit (part of Salesforce’s Composable Storefront) 12
- Managed Runtime (part of Salesforce’s Composable Storefront) 13
- In business terms, then, how is this Goldilocks? 14
- ‘I have proven the business case. Is this the only way?’ 14

Conclusion 15

Understanding Key Terms: Composable and Headless

What's the difference between composable and headless?

Let's start with two fast definitions.

Headless is a technical term. In commerce, it simply describes the separation of the front-end user experience code from the back-end processing code. We should keep in mind that headless really means 'many front-end experiences (heads) with one set of backend operations.' Think social shopping, watches, mobile apps, and websites.

Composable Architecture is a much broader term. It describes the ability of each system (e.g., commerce engine, email marketing system, content management system, product information system) to be connected to the company ecosystem once and reused many times through an integration layer. If in the future you disconnect, say, the email marketing system once from the integration layer and plug in the new one, all the other systems remain (largely) unaffected.

Sadly, these terms are often used interchangeably or incorrectly, leading to significant confusion for companies about the right way to go with their architecture and commerce decisions. Let's examine what the main messages in the market are.

Positives Messages in the Market

There are several positive messages around. Let's examine these first:

Headless is your future architecture.

This is proclaimed as gospel by many in the commerce world. There's some truth here: Having your front-end experience separated from your back-end logic creates a lower risk of front-end change and a higher ability to change quickly.

Headless on its own does not give you higher page speeds and more conversions. This is a myth, and we'll see why later.

Composable is the only architecture for a forward-looking company.

You must 'go composable.' This requires some unpacking. Many businesses have built significant technical palaces, which are technically composable (as defined above) but have provided little to zero business benefit. No conversion uplift, little change of pace or agility, and often, higher costs.

Why does this happen? It happens because a technical project is undertaken, often with a zealous approach to architecture, without a strategy to create business value. Here are a few mistakes we often see:

1. **Best in class:** This kind of architecture often comes with a desire to buy or (even worse) build best-in-class systems for each part of the composable architecture. They go with the full, enterprise-grade PIM, CMS, Commerce, etc. This creates a lot of OPEX (license) and CAPEX (putting it together to start with). Those costs are not offset by the business gain, which can be created only through revenue growth (because costs increased, they were not taken out).

2. **Ongoing management:** With many systems comes more maintenance overhead, likely a higher number of partners and, therefore, higher overall maintenance costs. This again increases our OPEX and significantly increases TCO.

3. **Missing the point:** In building a technical palace, they missed the business point. They didn't critically assess where the greater business results would come from (let's challenge greater 'agility' into something the CFO can calculate) and answered theoretical requirements. Requirements like 'Well, in the future, I want to be able to swap out my email system simply.' How often do you need to do this? When you do it, do you not change to take advantage of new features that downstream systems will need to adapt to anyway? What is the reality of the 'ease' of swapping? Again, this is something the CFO can calculate.

We'll increase our conversion rate.

Lots of business cases are predicated on an increased conversion rate. In fairness, some of the moves toward these newer approaches have yielded conversion rate differences. We do want this to be an outcome as it's one of our most calculable benefits.

This is generally driven by two factors:

1. **Redesigned experiences:** Sometimes, it is not the technology. A separate head (front-end code) from the backend (logic and processing) on its own does not create any change to the conversion rate. If you redesign the experience, you can expect to adopt an experience much closer to what your market needs, and that should create a conversion rate uplift. This is especially true of concentrating on the mobile experience for customers, which is now pushing 70%-80% of traffic. (Side note: If your partner starts by showing desktop designs, you're in trouble.)

2. **Page speed from modern front-end technology:** The second aspect that drives conversion rate change is page speed. We know that faster page speeds increase conversions. It feels better, flows faster and keeps people engaged. Often this is produced by a very different technical change to headless. That change is moving to what is often called a Progressive Web Application (PWA). The major difference this makes is that we don't reload the entire page experience every time the user clicks; we just reload the part of the screen we need to. For example, the header and footer likely only load once for the entire session. This makes a huge difference for speed, which can make a difference in conversions (and Google rankings).

So, yes, we will increase the conversation rate if we take one of these two routes (or maybe both!). Make sure you test these assumptions.

Negative Messages in the Market

The market also gives several negative messages about these changes, which sow fear, uncertainty and doubt (FUD) about these projects.

Headless (or composable) is expensive.

We explored why this message appears in the market in the previous sections. It's when technical palaces are created with a lot of expensive software, and they do not meet any business needs or outcomes.

There's confusion about whether headless or composable (remember our terms) creates the expense, but the whole area gets labeled this way. We already covered composable technical palaces (see Best in class).

"Keeping the idea of going headless simple is the key to ROI. Concentration on the business outcomes and measurable outcomes drives the right change."

It is also true that there are some very expensive ways to go headless. You could build from scratch your own bespoke head, assembling technologies like React or Next.js, GraphQL and many others into a technically amazing setup — and reinventing the wheel at the same time. Keep in mind that it is possible to make a headless setup slower than your current website if not done correctly. That likelihood increases the more you build from scratch due to the number of decisions that could be implemented incorrectly.

Another reason is linked to people getting carried away. One decision snowballs into another, and the whole thing gets super expensive. Here's an example: We want to go headless for the conversion rate benefits, so we'll choose a new front-end technology (PWA), and we'll redesign. So far, so good. Then, someone says they'd like much better content creation tools. You're not a very content-driven brand and have a very small content team, but the requirement isn't challenged. So, you now go to market and spend many cycles selecting a separate CMS.

The headless project now has more licenses and a slightly greater implementation cost. Your ongoing maintenance and use of the solution is split across your commerce and CMS solutions. Will this separate CMS really create enough business benefit for the significant expense? Depending on the business, it might, but in this example, where content is not huge for this business, it likely will not. We must challenge our assumptions.

Keeping the idea of going headless simple is the key to ROI. Concentration on the business outcomes and measurable outcomes drives the right change.

Headless (or composable) takes too long.

Headless/composable projects may take much longer than expected.
The market has seen this (and this is to our benefit, as you will see later).
What is disappointing is that this does not need to be the case.

The reason projects take longer is 100% aligned with the last section about why they are also more expensive: snowballing ideas, customized setups, and building technical palaces without business benefits. Let's say no more about this!

You need a deep technical team to maintain headless/composable.

This is only true if we've made all the previous mistakes. Deep technical teams are needed if you have a deep technical solution. For some businesses, there may be great benefits to this. If you're an F1 team, Uber or Amazon, technical competence and depth are key parts of your success as a product. For many commerce businesses, this is not necessary. Most commerce businesses are not a special case, where having a bespoke checkout is a differentiator in the marketplace. Most of these businesses differentiate by product range, pricing, availability, speed, brand and more — not technology.

If you've made the mistake of assembling a technical palace that does not drive business outcomes, you will have also assembled an unnecessarily large technical team to keep the lights on, more partner relationships and more software costs than needed.

Adopting A Balanced Approach: The "Goldilocks" Strategy

We have examined the positive and negative messages in the market. We've done some myth-busting and critical examination of the statements. But what do we do about it? How do we stop from becoming a headless composable chicken?

Start with the business

It may seem self-evident by now, but let's be clear: All technology projects should create a demonstrable, measurable business outcome or two. We should start with what we want to achieve and work with the technology to meet those needs. Concentration should be given to:

Revenue growth

- ✓ **Conversion rate:** Critically examine and prove how the new approach will achieve this and try and quantify it. Don't believe anyone who thinks a technical change will produce a 50% conversion rate increase. This is more often driven by redesign or page speed than any composable architecture

- ✓ **Speed of change or agility:** Often cited as a benefit of headless or composable is the speed of change. Immediately break this down. How often do you ship change now? What is the benefit of that change over time? If you could change two times as fast, what would that really look like? Could you release two times faster? Does your team, automated testing, and more allow this? Does your culture allow this? Ask the hard questions now. Don't take this benefit at face value.

Cost management

- ✓ **Rising costs:** When the business case is put together, are costs going up significantly with this change? If so, how does the revenue growth outstrip these increased costs?

- ✓ **Optimizing costs:** The ideal project can always take cost out. Can we automate our business processes more to get products to market faster? Can we spend less time per change? Can the team be smaller as a result? As before, critically assess the benefits and how they relate to cost.

"All technology projects should create a demonstrable, measurable business outcome or two."

We've examined the business case, the market trends, and myths. So, what should you do now? You should prioritize your business needs first and match the solution to those needs and outcomes. Here's how Salesforce matches some of the common issues with these projects. Part of the solution from Salesforce is **Composable Storefront**, which is made up of two components: the reference application and the hosting of that application.

PWA Kit (part of Salesforce's Composable Storefront)

This is a pre-built reference application built to use with Salesforce Commerce Cloud. It is a PWA, meaning if you don't already use this kind of technology (e.g., previous versions of Commerce Cloud like SiteGenesis or SFRA), then you get those page speed benefits that drive many business cases. **Here's the demo storefront**. It's fast and already has a home page, PLP, PDP, Basket, Checkout, My Account, etc. Don't reinvent the wheel. Apply your brand and differentiation for fast time to value. This saves a huge number of resources versus building from scratch (often called a bespoke head) and gives you the business benefits.

Managed Runtime
(part of Salesforce's Composable Storefront)

This is Salesforce's hosting environment for your new 'head,' built with the PWA Kit. Included in your license cost, you get Salesforce's famous uptime and speed without additional costs by using a third party. Build a bespoke head, and you also need to host that head (in AWS, GCP, Azure, Vercel, etc.) and carry daily costs for that, both in usage costs and the people to maintain and support it. Or let Salesforce do that for you.

Does using this Composable Storefront restrict me?
Maybe. It's not a bull answer. Restrictions are only restrictive if they make a difference to your business. Composable Storefront makes a set of decisions for you. If you're not happy with them, they can seem restrictive. If they don't matter to your business, they are an advantage and things you don't need to think about. Here are the main ones:

1. **It's built using React:** If you don't want to use React, then it might not be for you. I would challenge you, though. What business benefit will you get from using a different technology like Next.js that will be more beneficial than the time-to-market advantage of the pre-built Salesforce solution?

2. **It uses Chakra for UI:** Same thing as React. Maybe you want to use something different, but for what business benefit?

3. **Managed runtime flexibility:** There are some things managed runtime can't do, but what business benefit will your solution drive?

Some of you won't know what React, Chakra, etc., are. That doesn't matter. That's almost the point. They are technical choices already made, which could be seen as restrictions or could be an acceleration.

In business terms, then, how is this Goldilocks?

We have seen the challenges of headless and composable and the confusion around these terms. The use of Composable Storefront and Salesforce Commerce Cloud gives the following:

- A PWA that will drive page speed benefits (and, therefore, potential conversion and SEO benefits), leading to revenue growth
- A PWA built on popular technology (React), making the availability of a developed resource good if you do want an internal team
- A hosting of that PWA that adds no business liability on your side, leading to no additional tech team or maintenance
- A hosting of that PWA that is included in your license for no cost uplift
- No additional load on your team with many different systems to drive one website; Commerce Cloud still has competitive search, merch and content tooling that suits many customers, OOTB

If your project is focused on going headless, you can achieve it with Salesforce by getting the benefits of revenue growth without the increased cost the market often perceives with these projects. That's why it's Goldilocks. It suits the business and delivers outcomes in a world of frequent failure.

'I have proven the business case. Is this the only way?'

There's one great benefit of Salesforce's approach that we haven't mentioned yet. It does not close off any options for the future.

Let's say that your business does need a bespoke head. It needs to bring together the right PIM, the right CMS, and the right search/merch tooling in a composable approach (which some of our customers do).

Salesforce is a fully API-powered commerce engine running some of the top brands that use this approach without Composable Storefront. If the business benefits are there, this route can be the right one, and Salesforce fuels that just as well.

Conclusion

We said it once, and we'll say it again: Concentrate on business outcomes. Break down the revenue growth and cost implications. Prove the value. Select the right technology to match those needs. If a smart business stakeholder can't understand it, they shouldn't sign off on it. For many, the Goldilocks approach will power these business benefits without the technical downsides.

OSF Digital empowers businesses to implement the 'Goldilocks strategy' in headless and composable projects. With deep expertise in Salesforce technology and a focus on business-driven outcomes, OSF Digital offers tailored guidance to ensure that technology investments align with revenue growth and operational efficiency.

OSF DIGITAL

Ready to achieve a strategic, scalable setup that supports long-term success while adapting quickly to market changes?

Contact Us

OSF Digital is a global leader in digital transformation, specializing in Salesforce solutions that drive operational efficiency and business growth. With expertise in AI and composable architectures, OSF Digital empowers businesses to create seamless, future-ready customer experiences. Leveraging data-driven insights, OSF helps clients enhance performance, optimize processes, and scale for success. From innovative commerce solutions to managed services, OSF Digital is committed to helping companies maximize their digital investments and achieve measurable business outcomes.

THANK YOU FOR READING

Connect with us

osf.digital