

If you're weighing your options between **Netguru**, **DEPT**, and **Codal** for a composable commerce project, you're not alone. Composable commerce demands a sharp focus on modularity, clear ownership, and sustainable evolution—qualities that don't always show up in sales pitches or marketing decks. In this post, I'll break down these three agencies with a critical eye, focusing on what really matters: *cost control through disciplined scope, long-term operational ownership, system replaceability, and API-first architecture*.

The Stakes: Why Composable Commerce Isn't a 'Simple' Rebuild

Let's get real: composable commerce isn't just a tech upgrade. It's a fundamental shift in how your commerce stack fits together. Instead of one monolith, you build a collection of best-of-breed components united by *headless storefronts* and *API-driven integrations*. This approach offers flexibility, but only if you nail the architectural discipline.

Too many teams fall into the trap of "buy everything" or "build everything" then realize six months later they can't untangle their own stack. When I see a project buzzword like "composable commerce," my first questions are:

- Who owns this system *after* launch?
- How tightly are the system boundaries drawn?
- Can components be swapped out without a rewrite?
- Is there an API-first mindset guiding integrations?
- Is scope discipline enforced to control costs?

Let's look at how Netguru, DEPT, and Codal stack up.

Netguru: Strong in Long-Term Partnership and Modular Scope Discipline

Netguru has made a name for itself in mid-market commerce projects that prioritize flexible, modular scope. Their teams push hard on defining clear component boundaries and enforcing API standards right from day one. That's a huge plus for companies that want to avoid scope creep and layoffs-to-fix-technical-debt a year later.

Benefits of picking Netguru:

- **Modular Scope Discipline:** Netguru helps clients break larger Commerce as a Platform (CaaP) visions into manageable, replaceable modules. They keep a tight rein on scope, helping avoid "scope bloat" that kills timelines and budgets.
- **Long-Term Ownership Focus:** They emphasize documentation, developer enablement, and system maintenance from the earliest sprint planning. They ask "who owns this in year two?" at every checkpoint, which keeps expectations realistic.
- **API-First Architecture:** Netguru's engineering teams aren't content to just glue together existing APIs. They advocate evolving APIs thoughtfully, maintaining backward compatibility, and ensuring components can be safely swapped out without ripple effects.

Where Netguru might feel less comfortable:

- Less suited for short, one-off delivery projects. Their sweet spot is ongoing partnerships.
- Smaller operational footprint, so projects requiring complex multi-vendor orchestration might need additional vendor support.

DEPT: Broad Capabilities but Beware of Over-Engineering

DEPT is a powerhouse that brings broad digital marketing and technology expertise. They can definitely build you a headless storefront with API-driven integrations. However, their breadth sometimes leads to wide-open project scopes where every new shiny integration or feature [ecommerce replatforming guide](#) is “possible”—which can derail composable commerce’s promise of controlled scope.

What DEPT offers:

- **Comprehensive Service Breadth:** From UX to Martech to engineering, DEPT can staff large-scale, end-to-end commerce projects that span marketing and operations.
- **Innovative Approach:** They’re early adopters of composable concepts and push creative technology stacks.

But where DEPT stumbles in composition:

- **Risk of Over-Engineering:** DEPT’s “we can do anything” pitch can backfire when it leads to uncontrolled scope or stacking new vendors for every new feature, which adds complexity.
- **Less Clear System Boundaries:** Because they cover such broad service areas, sometimes the lines between components get blurry. This can hamper replaceability and increase long-term operational costs.
- **Ownership Ambiguity:** It’s common to hear “let’s bring in the new team” after delivery, which is a red flag for any composable project that needs continuity.

Codal: User-Centered Design Meets API-Driven Execution (But Watch the Handoff)

Codal is known for user experience and design-first thinking, with solid chops in API-driven commerce stacks. Their focus on user interfaces meshes well with headless storefront needs, and they do a credible job with connectors and integrations.

Strong points for Codal:

- **UX-Driven Architecture:** They build storefronts and integrations designed for user-centric flexibility and responsiveness.
- **API-Driven Development Culture:** Developers and designers collaborate closely to ensure API contracts enable flexible frontend experiences.
- **Agile Delivery Models:** Codal tends to deliver quicker results for well-scoped, focused projects.

Caveats when working with Codal:

- **Long-Term Support Risks:** Codal is great for initial delivery, but ongoing operational ownership may need internal teams or additional vendors.
- **Limited Enterprise Integrations:** Compared to Netguru or DEPT, Codal sometimes leans on smaller ecosystems, which can complicate replaceability if you scale quickly.

Table: Quick Comparison Summary

Criteria	Netguru	DEPT	Codal	Scope	Control	Disciplined	modular scope, avoids scope creep
Broad but sometimes prone to scope bloat	Focused scope, user-centric features	Long-Term Ownership	Emphasizes operational ownership and documentation	Often handoff after delivery, less continuity	Good initial delivery, less clear on operational support	API-First Design	Strong API-driven architecture with controlled evolution
API friendly but							

sometimes wide integration networks API-driven UX focus; good collaboration Replaceability Clear system boundaries enable easy component swaps Component boundaries can blur in large projects Good for front-end replaceability; backend less flexible Best For Long-term partnerships, mid-market via modular composable stacks Large, innovative projects needing broad marketing + tech UX-heavy projects needing fast, focused deliveries

Netguru vs DEPT vs Codal – The Final Word

Netguru vs DEPT



If your key priority is control—especially cost control—and long-term ownership of your composable commerce ecosystem, Netguru edges DEPT. DEPT’s “we can do anything” promise sounds nice until you’re managing a complex network of vendors and integrations without clear ownership or boundaries. Netguru’s disciplined modular approach builds from the ground up with scope control and system replaceability baked in.

DEPT vs Codal

DEPT is better if you want an integrated approach spanning digital marketing, engineering, and innovation. Codal is a smart pick if your focus is primarily on front-end user experience and quick delivery of headless storefronts—but be ready to reinforce backend integration and operational ownership either internally or with another partner.

Netguru vs Codal

Netguru shines on long-term evolution and clearly defined system boundaries in composable architectures, particularly when your team plans to manage the platform beyond launch. Codal plays well for sleek frontends and user-focused API design but can leave gaps in operational continuity after project completion.

Cost Control Through Modular Scope Discipline

One of the biggest hidden costs in composable commerce is scope creep. The temptation to add features, integrations, or vendors at every turn leads to ballooning timelines and budgets. Agencies that understand this (hello, Netguru) insist on strict modular discipline—building chunk-sized modules, testing them end-to-end, and only then adding the next piece. This approach not only protects budgets but also simplifies troubleshooting, upgrades, and component replacements down the line.

Long-Term Ownership vs One-Off Delivery

Anyone can build and launch a headless storefront or weave together APIs. The harder question is: who owns it in year two? Is that system going to rot, or does someone have skin in the game to keep APIs evolving, documentation current, and dependencies managed? Netguru's model centers on partnership beyond launch, while DEPT and Codal often shine at delivery rather than ongoing ops.

Clear System Boundaries and Replaceability

Composable commerce's promise is the ability to swap a single component without breaking everything else. This requires razor-sharp system boundaries and contracts. Netguru's architects invest time upfront ensuring each piece is replaceable behind clear API interfaces. DEPT's sprawling service offerings sometimes muddy these boundaries, and Codal's focus on frontend UX means backend replaceability may take a backseat.

API-First Architecture and Controlled Evolution

APIs are the glue holding your composable stack together. But sloppy API design leads to brittle integrations and roadmap chaos. Agencies experienced with composable commerce treat APIs as first-class products, planning for backward compatibility, versioning, and evolution. Netguru's engineering teams excel here, crafting APIs that enable controlled, incremental system evolution—vital for long-term resilience.

In Conclusion

Choosing between **Netguru vs DEPT vs Codal** isn't about picking the biggest agency or the flashiest portfolio. It's about matching your project's maturity, scope discipline needs, and operational mindset with a partner who understands composable commerce is a marathon, not a sprint.

If cost control, API-first modularity, and sustainable long-term ownership top your priorities, start your conversations with **Netguru**. If you want expansive digital capabilities but beware scope bloat, **DEPT** might fit. If your focus is on quick, user-centered headless storefronts and you have internal resources to support backend evolution, **Codal** is worth exploring.

Whatever your choice, challenge vendors early with blunt questions:

- Who owns this in year two?
- How do you enforce scope limits?
- Can components be replaced without a full rewrite?
- How do you manage API versioning and backward compatibility?

Composable commerce is complex but rewarding. Choose a partner who sees the end game, not just the launch day.

