

The Lego-fication of Finance^{*}

Why Composability Is the Most Important Word You're Not Talking About

Siddhant Shah 18 Feb 2026 7 min

Composability

DeFi

THE SHORT VERSION

There is a word that gets thrown around in blockchain circles constantly, occasionally surfaces in AI discussions, and almost never gets the serious treatment it deserves in mainstream finance. That word is **composability** – and if you work anywhere at the intersection of finance and technology, understanding it deeply might be the single most valuable thing you do this year.

I've become convinced it's not just a technical property of blockchain systems. It's the organizing principle of where finance is actually heading. And it's arriving on two fronts at once.

What Composability Actually Means (Without the Jargon)

Composability is the property of a system where individual components are designed to work with each other – so you can freely combine them to create new products without rebuilding from scratch each time.

That definition is accurate and also kind of useless. Here's the version I actually use.

Imagine the entire universe is made out of Lego (yay). Not that Lego exists in the universe – that the *universe itself* is Lego. Every capability, every service, every piece of financial infrastructure is a brick. And because every brick is built to the same standard, every piece fits with every other piece. You never ask “will this connect?” The only question is: what are you trying to build?

If you've seen *The Lego Movie*, you know the concept of a Master Builder – someone who looks at a pile of pieces and doesn't see what they are, but what they *could become*. Here's the thing about that framing that I keep coming back to: a Master Builder isn't necessarily a programmer. It's anyone – an analyst, a portfolio manager, a product lead, a founder – who understands the components well enough to see possibilities hiding inside them. That's the skill composability rewards. Not just execution, but *vision*. The ability to see the product before the pieces have been assembled.



The Wall Street Bull, made of Golden & Blue Steel Lego Bricks.

Blockchain: Where the Lego Bricks Got Laid First

The crypto world has been living with composability for years, and Ethereum is the clearest proof of what it actually unlocks.

Every protocol built on Ethereum is designed to interact with every other. A developer building something new doesn't start from zero – they call on Uniswap for liquidity, Aave for lending, Chainlink for price data, and stitch them together into something novel. What would have taken months of engineering can be assembled in days, because the pieces already exist and they fit. Every new protocol that ships becomes a potential brick for future builders. The creative surface area of the ecosystem grows with every launch.

This is what the “DeFi Legos” framing in crypto is actually pointing at. Ethereum's composability is one of the core reasons it dominates – not just as a network, but as a *development environment*. It doesn't just enable building; it accelerates it compounding.

But here's the piece of this that doesn't get enough attention: **composability in blockchain doesn't just enable collaboration – it solves the attribution problem.**

Think about what happens in traditional software when you build on someone else's work. Open source runs on goodwill and reputation. You integrate a library, use an API, and the original author gets... maybe a GitHub star. That's been the accepted reality for decades, and it carries a quiet injustice – foundational builders often go uncompensated for the leverage they hand to everyone who comes after them.

Smart contracts change this mechanically. When a protocol is built with proper incentive design, every time another protocol uses it – every time someone routes a trade through it, borrows against its liquidity, or calls its functions in a composed product – fees flow automatically back to the original creators. The attribution isn't moral or

reputational; it's enforced by code. You can literally program the recognition of foundational work into the infrastructure itself.

This matters beyond the fairness argument. It means builders at every layer of the stack have a genuine *financial* incentive to make their work composable and accessible. Composability stops being a design philosophy and becomes an economic strategy. That changes how ecosystems develop.

The Standardization Connection: This Isn't New, Actually

Here's something that clicked for me when I started thinking about this more carefully: composability isn't a blockchain invention. It's what you naturally get when standardization matures – and finance has been pursuing standardization for a long time.

We standardized financial statements so any investor could pick up any company's filings and immediately know where to find what they needed. We standardized settlement processes, cleared trade structures, reporting formats. Every time the industry agreed on a common way to represent something, it got easier to work with and build on top of. Technology followed the same arc – TCP/IP standardized how data moves, REST APIs standardized how applications talk to each other, container formats standardized how software deploys.

At each step, standardization abstracted away implementation details and let builders focus on higher-order questions. Not *how* does this work, but *what do I want to create with it*. The shift from implementation to orchestration. That's what composability enables – and that's what finance is now reaching across its entire technology stack.

AI: The Second Wave, Same Pattern

While blockchain has been building composable infrastructure for years, AI is going through the exact same transition right now. Watching it happen in real time has honestly made me more bullish on the composability thesis overall, because the pattern is identical.

Early AI in financial services was a series of isolated black boxes. Fraud detection here. Credit scoring there. Sentiment analysis bolted onto a research workflow. Powerful in isolation, siloed in practice. One model's output didn't naturally feed another's input. Connecting them required custom engineering work every time. The pieces didn't fit.

That's changing on a few fronts simultaneously.

LLM orchestration is chaining language models together – routing tasks to the right model based on what it's good at, passing outputs from one as inputs to another. An investment research workflow might use one model to extract structured data from earnings transcripts, pass that to a model optimized for financial analysis, and route the output to a third that formats it for distribution. Each model is a brick. The orchestration layer is the assembly.

DSPy takes the composability idea and applies it to prompting itself. Instead of hand-tuning prompts to a specific model – which breaks the moment you want to switch providers or a new model ships – DSPy lets you define the *logic* of what you're trying to accomplish: the reasoning steps, the input-output structure. It optimizes the actual

prompts automatically. Your workflow becomes portable. When a better model comes along, you swap the component and the framework adapts. Prompt engineering, composable.

MCP (Model Context Protocol) servers extend this into tooling. They create a standardized way for AI models to interact with external tools, data sources, and systems — databases, APIs, internal platforms, file systems. Instead of custom integration code every time you want an AI to touch a new data source, you connect an MCP server and the model uses it immediately. The ecosystem of tools becomes composable with the ecosystem of models.

Same pattern. Different layer.

What This Looks Like in a Finance Workflow

Here's where it gets interesting. Take a **morning briefing pipeline** for a portfolio manager: an MCP-connected market data tool pulls overnight moves → a fast language model summarizes relevant news → a financial analysis model flags portfolio-relevant developments → a formatting layer produces the brief. Each component built once, reusable across clients, asset classes, formats. Add a new data source? Slot in another brick. Change your preferred model? Swap it without touching the rest of the pipeline.

Or **document analysis for deal teams** — a stack of offering memos where you need key metrics extracted, benchmarked against comparables, and formatted into a one-pager. Extraction logic, benchmarking logic, and formatting logic are separate reusable components. Need to add ESG scoring? There's a brick for that.

Or **client communication at scale** — personalized portfolio commentary using a model-agnostic prompting framework, so reasoning and guardrails stay consistent while the underlying model can flex based on cost and latency requirements. The logic travels. The implementation details don't matter.

In each case, the value isn't in any single model or tool. It's in the assembly. And this is why the Master Builder framing matters beyond engineering — a portfolio manager who understands what an LLM is good at, knows what data sources are accessible, and can sketch out how pieces connect can design something that would have required a full engineering team two years ago. The bottleneck shifts from technical capability to vision.

We Are at the Beginning of Something

The financial industry has always moved toward greater standardization — not because it's aesthetically appealing, but because it lowers costs, reduces friction, and expands the universe of possible transactions. Every financial innovation that's scaled has done it by making something composable: the bond market standardized debt instruments, derivatives standardized risk transfer, index funds standardized portfolio construction.

Blockchain and AI are the current frontier of this centuries-old process. They're bringing composability to the infrastructure layer of finance itself — to how value moves, how decisions get made, and how financial products get built.

The honest caveat: composability only goes as far as standardization actually holds. Finance has a history of near-misses where interoperability almost emerged and then got captured by incumbents with no incentive to let the pieces fit. Whether that happens here is a real question — I just think the economic incentives in this cycle are structured differently enough that the outcome is too.

The pieces are being laid. The standards are being written. The early builders are figuring out what fits. The people who develop an eye for composability now – who can look at a pile of protocols, models, and tools and see the product hiding inside them – are going to have a real edge in what gets built next.

The Master Builders are just getting started.

DISCLOSURE

The views expressed here are my own personal opinions. This is **not investment advice** and should not be relied upon as such. Nothing here is connected to, endorsed by, or written on behalf of Rosenblatt Securities.

© 2026 Siddhant Shah