Anoma is unbundling technology, literally.

Note to the reader

: this is a very raw version of a first draft which the author intends to complete over the coming week. One of the challenges is trying to distill some of the core concepts into a digestable form for a builder, researcher audience while remaining above low level details while allowing for light-hearted philosophical musings. Feel free to complain if you think the structure or direction of this art is off-the rails or has somehow found its self in the gutter.

There have been many attempts to explain Anoma over the last few years. Explanations span from a unified architecture for full-stack decentralized applications

[1]

to a tool for coordination

[2]

, and further to contemporary explanations such as, Anoma as the universal intent machine for Ethereum

[3]

.

The Anoma vision paper laid out something else, Anoma: Undefining Money

(versatile commitments to value) [4]

. While the versatile commitments to value is quite vague, undefining money pokes at a concept and framing the author thinks best describes the vision of what Anoma is.

Anoma is unbundling technology

.

Unbundling is a concept often used in the discourse to describedisruptive technology. In particular, by unbundling we mean breaking up things that were previously offered in a group [5]

. Though its not enough to simply unbundle things without a proper motivation or reason less of course it brings the unbundler child like joy or some other emotional fulfillment. Therefore, we'll not only discuss four examples of how Anoma unbundles technology, taking inspiration from Virgil Griffith [6]

, we'll also explain the motivations or raison d'être.

Example 1: Anoma unbundles protocol architecture from network topology with intents

Network architecture is a framework for the specification of a network's physical components and their functional organization and configuration, its operational principles and procedures, as well as communication protocols used.

Network topology, on the other hand, is the arrangement of the elements (links, nodes, etc.) of a communication network.

Let's break this down by way of a thought experiment. Imagine you are designing a city. The citiy's architecture is the high-level design for the city, including where everything is supposed to be and how it all works together. The city's topology would be how you decide to connect everything, roads, bridges, parks, sidewalks, etc. The topology determines the best way for people to move around the city.

More concretely, Ethereum's Architecture is like the protocol stack. It defines core components like EVM, accounts, transactions, blocks, and consensus. The architecture specifies high-level system design principles and interactions which includes features like smart contract languages, EVM upgrades, and Ethereum improvement proposals (EIPs).

Ethereum's topology is how it works in practice. In particular, (i)

it describes the organization and interconnections of different node types, (ii)

outlines roles like full nodes, light nodes, miners/validators, boot nodes, (iii)

illustrates flow of data/transactions between nodes, and (iv)

is designed to support core functions like transaction propagation, consensus, execution. In simple terms, you can think of

Ethereum's topology as the MEV supply chain, which is continuing to evolve [7]

, signaling signs of a healthy network[8]

.

What does architecture and topology have to do with intents? Recall that intents are

- colloquial definition: credible commitments to a preference function over a shared state space
- mathematically: atomic information flow constraints

Precisely:

$I := (S \to S) \times (S \times S \to U^\star [0,1])$

Intents, I

are formulated as a pair, consisting of a transition function

$(S \to S)$

and a partially weighted predicate

over state transitions $(S \times S \to U^\star [0,1])$

.

The guiding intuition for this formulation is to separate (unbundle) control from desire[9]

. In the prototypical example, intents will express a desire for some resource in exchange for another. The 1st component expresses a partial state transition, where the intent may create/destroy what it has control over. This aids in composing intents. The 2nd component expresses a weighted predicate over transitions. If the transition satisfies the intent, it returns an element between 0 & 1, representing a kind of utility.

Intent-centric architectures like Anoma provide a specific way of organizing subcomponents into a structure designed to serve the purpose of the overall system. Two relevant examples are p2p routing and transaction execution.

- Traditional P2P routing architectures have P2P routing hardcoded into the core protocol, see the Tendermint Mempool as an example. Alternatively, Intent-centric architectures separate P2P routing into subcomponents which allow different implementations prioritizing speed, programmable disclosure, and trust.
- Traditionally, transaction execution is tightly coupled with consensus (agreement on who owns what) and data storage. On the other hand, Anoma decouples counterparty discovery and settlement. This enables different execution environments (EVM, SVM, Move, ARM…).

Anoma's intent-centric architecture allows for the creation of a diverse set of topologies.

- Roles and responsibilities explicitly defined based on intents.
- Programmable node configurations with dedicated nodes for routing, solving, execution, consensus, etc. - all agents run Anoma nodes configured at runtime for their desired roles.
- Promotes flexibility for different users/communities to experiment with suitable topologies.

Indeed, Anoma unbundles protocol architecture from network topology with intents, which affords greater flexibility for users and developers with varying degrees of preference expression.

Future Work

Our work is only partially complete. In examples 2-4 we'll discuss how:

- Anoma unbundles protocols from operators
- Anoma unbundles tokens from protocols
- Anoma unbundles money from convention
- Adrian Brink, Awa Sun Yin, Christopher Goes, [Anoma: a unified architecture for full-stack decentralized applications

](https://github.com/anoma/whitepaper/blob/main/whitepaper.pdf), 2022. ↩

1. Christopher Goes, [Anoma: Endgame - Anoma Huddle Amsterdam

](https://www.youtube.com/results?search_query=Anoma+endgame), 2022. ↩

1. Christopher Goes, [[RFC] [DRAFT] Anoma as the universal intent machine for Ethereum

](https://ethresear.ch/t/rfc-draft-anoma-as-the-universal-intent-machine-for-ethereum/19109), 2024. ↩

1. Christopher Goes, Awa Sun Yin, Adrian Brink, Anoma: Undefining Money, 2021. ↩
2. Wikipedia contributors, [“Unbundling,” Wikipedia, The Free Encyclopedia

](https://en.wikipedia.org/w/index.php?title=Unbundling&oldid=1198789735), 2024. ↩

1. Virgil Griffith, [Ethereum is game-changing technology, literally.

](https://medium.com/@virgilgr/ethereum-is-game-changing-technology-literally-d67e01a01cf8), 2019. ↩

1. Frontier Research, [Infinite Games

](https://frontier.tech/infinite-games), 2024. ↩

1. Christopher Goes, [Towards an intent-centric topology

](https://anoma.net/blog/towards-an-intent-centric-topology), 2024. ↩

1. Anthony Hart, D. Reusche, [Intent Machines

](https://zenodo.org/records/10654543), Anoma Research Topics, 2024. ↩