Exploring the Use Cases of Cheap Computation

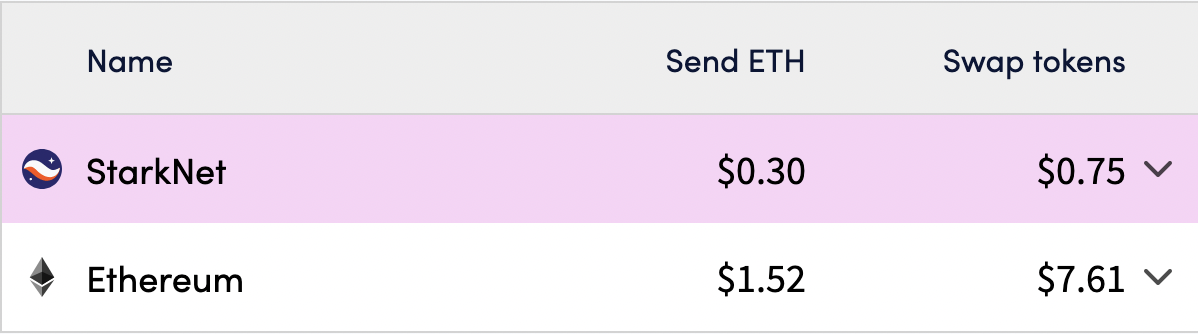
*What new use cases does Starknet enable, how can devs take advantage*

In the bull markets of 2021, when everyone was talking about crypto and sending out millions of transactions on Ethereum, the gas price used to reach 100 Gwei easily, meaning if you wanted to send 10 USDC to someone on Ethereum, that would cost you 10-15 $. Bluntly, not a very fair deal.

What happened was that with many new users of the blockchains, the demand for block space skyrocketed, which the chains were not yet ready to support. This surge in gas prices on Ethereum (and other chains) exposed the limitations of monolithic blockchains. To address this issue, a shift to modular design emerged with “layer-2” solutions, a set of technologies built on top of a base blockchain that increases scalability and cost-effectiveness. As per Polynya’s [article](https://polynya.mirror.xyz/wnt16H0VnFoCGDK_ds5H4J1ZcHepZfQr1DQHCzw940o), “all blockchains worth their salt will pivot away from purely monolithic, in some way or another, over the coming years (unless the scale is not required, like Bitcoin.).”

On Ethereum, these L2s are available in the form of optimistic rollups ([link](https://ethereum.org/en/developers/docs/scaling/optimistic-rollups/)), zero-knowledge ([link](https://ethereum.org/en/developers/docs/scaling/zk-rollups/)) rollups, and state channels ([link](https://ethereum.org/en/developers/docs/scaling/state-channels/)). The magnificent thing about rollups is that they derive their security from Ethereum L1 (thus utilizing a decentralized network of over [600k validators](https://beaconcha.in/validators)). However, they also offer much faster and cheaper execution than an ordinary transaction on the mainnet. They achieve this by performing the computations outside of layer one but periodically posting the state of the L2 on the mainnet as a bundle (hence “rolling up” the transactions). The cost of posting the updated state (a commitment) is divided between the users who sent the transaction on the L2s. An apparent outcome is that the cost of sending transactions on rollups increases as the demand for block space increases (did we break the demand-price curve of economics).

For example, Starknet is a validity rollup (commonly known as a zero-knowledge rollup) that uses cryptographic systems called STARKS to achieve decreased computation cost. The [StarkWare](https://starkware.co/starknet/) team, an Israeli-based blockchain development company, develops it. Since Starknet is a rollup, the computation cost is expected to go down with time (moreover, the cost could go down more than 10x after the introduction of EIP-4844). Nevertheless, even today, the mainnet version of Starknet already has transaction fees much lower than Ethereum.



Source : <https://l2fees.info/>

As blockchain technology continues to evolve and expand, the demand for complex computations will only increase, leading to higher user costs and potential limitations on the growth and adoption of decentralized applications. Even today, many applications are limited in their use cases due to this high computation cost.

1. Day-to-day transactions. The promise of “crypto as cash” never actually came true (except for [some](https://www.pwc.com/gx/en/financial-services/pdf/el-salvadors-law-a-meaningful-test-for-bitcoin.pdf) exceptions where you can “survive on crypto”). Along with regulatory uncertainty, one of the primary reasons for this has been expensive transactions. Nobody would pay 1$ in transaction fees for a coffee worth 2$.
2. Oracles. Due to their inherent design and principles, blockchains cannot retrieve off-chain real-life data. In order to get such data as stock prices, weather data, sports match results, etc., special applications called oracles are used. These oracles periodically publish data on-chain, which smart contracts can retrieve. However, the major problem is that due to the high cost of transactions, the oracles cannot publish data on-chain very regularly and often have large intervals. Resultantly, the data on-chain becomes outdated and might cause unruly decisions made by smart contracts. Moreover, due to expensive execution, any kind of data aggregation or additional computation becomes unfeasible.
3. Governance. On-chain governance, or on-chain voting, is an effective way of making decision-making processes transparent and decentralized. Until now, for on-chain voting to occur, the voters were required to cast their vote (based on the amount of token X they hold or any other mechanism to achieve voting power) by sending a transaction. However, for governance to be decentralized, effective decisions must be made very often and in a fast-paced manner. It became an inefficient voting system because people simply skipped their votes to save on transaction fees since the computation cost was so high. Consequently, many projects opt for off-chain governance, which undermines the purpose of blockchains.
4. Gaming. When we hear of on-chain gaming, the once-viral game [Axie Infinity](https://axieinfinity.com/) comes to mind. In the game developed by Sky Mavis, the game characters, Axies, were represented by NFTs. However, the complete game logic was running in a fashion similar to all the other games in a centralized server. Games require players to make numerous decisions in a short period, each representing a transaction of sorts. Moreover, based on the player's decision, the game plot changes accordingly, which requires large amounts of computation. This limitation of expensive computation caused games like CryptoKitties and Axie Infinity to feature only limited on-chain components. However, by marketing themselves as "web3" games, they achieved a market cap of several billion dollars at their peak, despite not being genuinely representative of blockchain gaming.

Now let us discuss what new use cases cheap computation enables. Starknet, a validity rollup, is already live on mainnet and processes thousands of transactions (with more than 7 million transactions done to date). Once EIP-4844 is integrated, the cost of transactions will go down multiple times. Let us assume that the EIP will be successfully integrated in the near future and that the computation cost has decreased even more.

Starknet enables new use cases that the developers and users must take advantage of. Let us take a look at some of them:

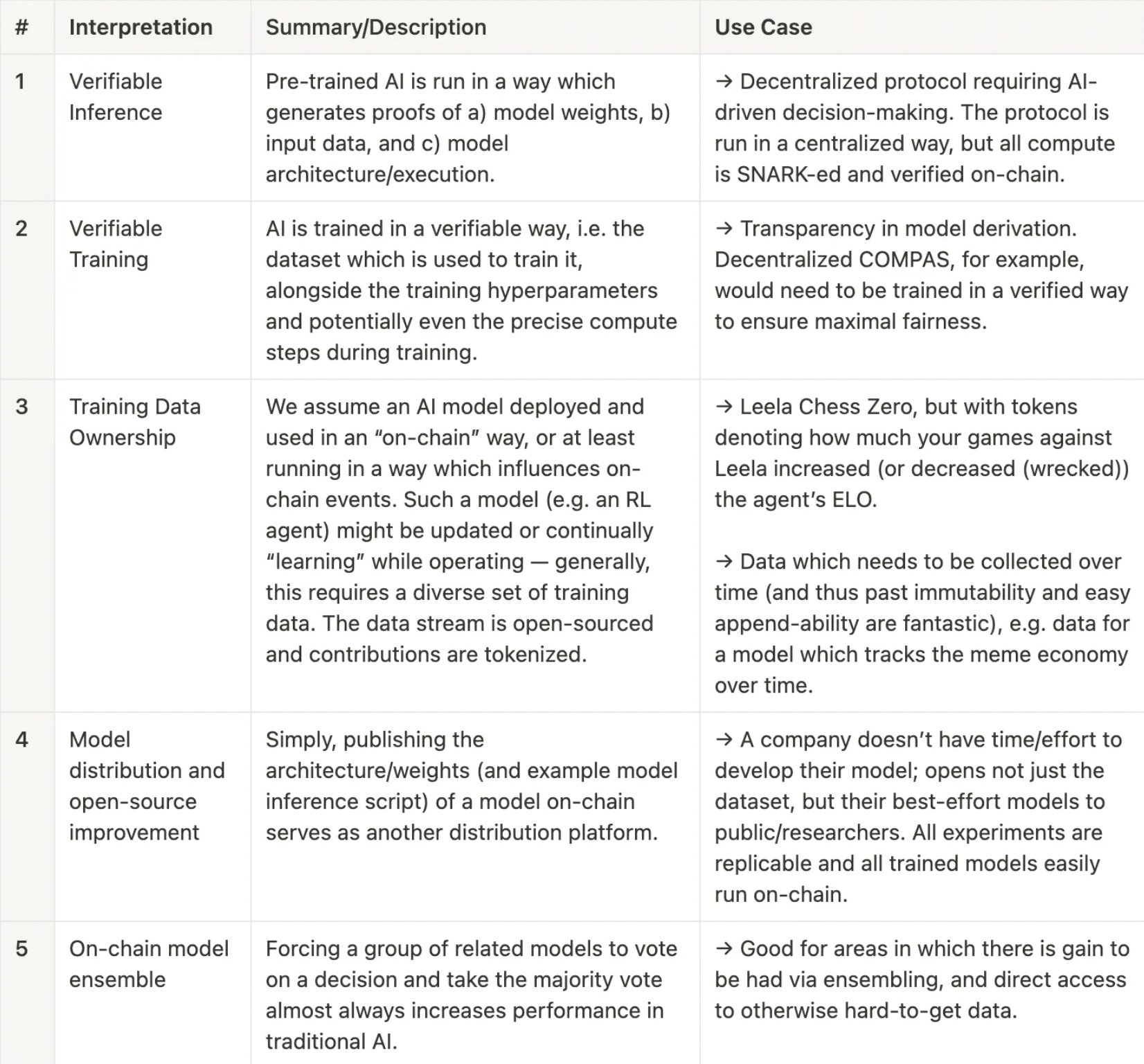
1. The simplest of them all is that with cheap transactions comes high liquidity. Paying for your coffee costs you a couple of cents (or even a fraction of a cent). Using crypto daily for simple payment transactions is made possible by low transaction costs. Additionally, due to account abstraction, your crypto wallet functions similarly to a traditional wallet. Much like keeping the bulk of your finances in a secure savings account, you store most of your assets in a wallet with multiple security layers (e.g., requiring a trusted individual to co-sign large transfers or using a unique private key).
2. With the introduction of cheap on-chain computation, oracles will look much more different than today.
   1. First, **frequent updates** of Oracle feeds become possible since individual transactions cost much less. This will result in more accurate data feeds and would enable more use cases for the users to build markets based on this data (complicated options protocols, prediction markets, etc.).
   2. Since Starknet is a rollup, it needs to periodically send a commitment on Ethereum mainnet to update the state of the rollup. This commitment is one of the significant costs that a user pays when paying transaction fees on an L2. However, the exciting part is that the rollup must only commit the latest state to L1. Overwriting the same storage slot multiple times in a single block will not incur high storage costs. The cost would still be equivalent to a single write as only the final state is published to mainnet as calldata.
   3. **Computational feeds** become possible due to the cheaper cost of execution, and teams like [Pragma](https://www.pragmaoracle.com/) (previously known as Empiric) are working on it. Pragma already has two computational feeds live on Starknet testnet - a volatility index and a yield curve.
   4. The utilization of **storage proofs** will also change the landscape of oracles. Storage proofs are a cryptographic way to keep track of storage. Using these cryptographic commitments makes it possible to trustlessly prove that a particular state existed at a certain time (or at a specific block, in the case of blockchains). In order to verify these proofs, computations are needed (though they are very light), and cheap computation will enhance the UX. With storage proofs, it becomes possible to transfer information between different chains in a decentralized manner, offer historical data from the blockchain, and much more. If you want to dive deeper into storage proofs, check out our Medium article [here](https://starkware.medium.com/what-are-storage-proofs-and-how-can-they-improve-oracles-e0379108720a).
3. The world of gaming is immense. The largest gaming market, the USA, caps a whopping **54 billion USD** per annum in revenue from gaming ([source](https://www.statista.com/forecasts/308454/gaming-revenue-countries)). Blockchain games have been in the spotlight recently, yet, as previously discussed, those games mainly belong to “web2.5” rather than actual “web3” games.
   1. In order for games to be actually considered a blockchain game, it must:

|  | Web 2 | Web 2.5 | Web 3 |
| --- | --- | --- | --- |
| Have the game-related data on-chain. | ❌ | ✅ | ✅ |
| Have the **logic of the game on-chain**. | ❌ | ❌ | ✅ |
| Be not reliant on the game devs to be everlasting | ❌ | ❌ | ✅ |

For the game logic to be completely on-chain, cheap execution is essential. For the game to be considered a web3 game, an elementary “litmus test indeed” could be whether the game could survive if the developers behind the game suddenly disappeared. Very similar to how a smart contract lives on-chain, no matter if the developer of the smart contract is still active or not. Different interfaces could be built on the game logic if the basic game mechanics and rules are on-chain. The developers, on the other hand, would be incentivized to build with game-specific tokens. Even though no major game is currently wholly built on-chain, multiple projects on Starknet are working in the right direction, including [Dark forest](https://zkga.me/), [Realm](https://linktr.ee/BibliothecaDAO), and [Influence](https://venturebeat.com/games/unstoppable-games-will-launch-web3-sci-fi-mmo-influence-on-starknet/).

* 1. Ecosystem developers like [Dojo](https://dojoengine.org/) are working on providing an open-source ECS (Entity-Component-System) framework for the Starknet ecosystem. Projects like such are needed as the ecosystem grows, and more game developers are joining to build games on Starknet.

1. With the recent buzz of artificial intelligence, it seems that 10 years down the lane, artificial intelligence will be integrated into every part of our digital being. However, why would we want to put AI models on on-chain? The short answer is to decentralize the models. The long answer is not so straightforward. It is so because of the nascency of both of these technologies. However, it would be beneficial to understand that there are multiple levels of bringing our AI models on-chain.
2. **Verifiable inference** means that models are pre-trained in centralized servers. However, once training is done, the model weights are published on-chain. Once this is done, inputs could be passed to the model on-chain, and the outputs would maintain transparency with the user having proof of the model generating a certain output.
3. **On-chain model ensembles** could be made possible if computation is made cheap enough for models to be run to generate outputs on-chain. Outputs from multiple models could then be aggregated to form an “ensemble” model commonly used in ML applications.
4. **Verifiable training** would mean that models could be trained on-chain, maintaining verifiable proof of benign training without introducing external biases. However, this may be the most compute-intensive application of the above but also the least likely to be brought live soon. These days, large AI/ML models take days (or years) to train on GPUs with enormous resources. On top of this, generating S[N/T]ARK proofs for the computation done while training would be an overhead right now. Moreover, Cairo specifically runs on CPUs (which are much slower than GPUs).



Source: <https://www.moduluslabs.xyz>

[Modulus Labs](https://www.moduluslabs.xyz/) (*which magically abbreviates to ML*) is one of the prime research teams working to bring AI on-chain using ZK proofs. There is certainly much innovation to be done in the space.

**What is the future for Starknet ?**

Starknet is still in a nascent stage of its future as a rollup. In order to understand the roadmap of Starknet, let us quickly go through how rollups function. When someone submits a transaction on the L2 solution, the first stop is the **sequencer** - a software responsible for ordering transactions. Once the sequencer creates a block, it forwards the block of transactions to the **prover**, responsible for creating a cryptographic proof for the ordered set of transactions (this proof is submitted on Ethereum L1, hence “inheriting” the security of Ethereum). Currently, the sequencer is a private software written in Python. Even though it solves the purpose it was created for, the StarkWare foundation has already started building a new open-source version of the sequencer in Rust which would be more performant.

Currently, the sequencer *sequentially* processes the transaction. However, a substantial performance improvement can be observed if the sequencer can process transactions in parallel. A more detailed example of how this is possible is included in [this](https://www.starknet.io/de/posts/engineering/starknet-performance-roadmap) article.

With the intensive research around the “modular blockchain” space, the “bringing one billion users” goal is getting closer and closer.