dYmension: Home of the RollApps

August 31, 2022
v1.0

Abstract

Present day blockchains operate as shared bandwidth systems which impede the growth of decentralized applications. dYmension disaggregates resource consumption by introducing a multi-layer blockchain protocol with robust tooling for building and deploying permission-less application-specific rollups. dYmension is akin to a hub and factory for such applications, which are referred to as *RollApps*. In this document we examine the technological, economic and social aspects of the dYmension protocol. Furthermore, we present the grand vision for dYmension, the modular architecture of its design and a high level overview of dYmension's prominent core concepts.

- **Rollup Factory - RollApp Development Kit (RDK):** dYmension's rollup factory takes its inspiration from the Cosmos ecosystem which introduced the successful Cosmos SDK. A RollApp instance on dYmension is an application-specific rollup, built using the dYmension RollApp Development Kit, termed RDK. The development kit is a pre-packaged set of generic modules for common functionalities such as creating accounts and token management. The RDK simplifies the process of deploying a RollApp on top of dYmension's settlement layer.
- **dYmension Settlement Layer:** A Cosmos SDK Proof-of-Stake chain which utilizes the Tendermint Core state replication model for networking and consensus. Contrary to a monolithic blockchain, dYmension's settlement layer, also referred to as a Hub, is specifically designed to provide an optimized service for rollups. As such, rollup servicing logic is enshrined within the settlement layer, resulting in a hub for native interoperability between RollApps.
- **Inter-Rollup Communication (IRC):** IRC is an Inter-Blockchain Communication (IBC) based protocol which provides safe message transferring between dYmension RollApps. IRC leverages the common communication ground of all dYmension RollApps, the dYmension settlement hub.

- **RollApp Virtual Machine (RVM):** dYmension introduces a novel dispute-resolution mechanism which simulates a RollApp execution environment within the settlement hub. The settlement hub spins up an RVM instance which is being fed with the exact context of a given transaction, resulting in a deterministic output. As such, dymension is capable of supporting various execution environments.
- **Embedded Hub AMM:** dYmension embeds a native Automated Market Maker (AMM) into the settlement hub to achieve shared liquidity on top of shared security. The AMM is designated for RollApp facilitation and is regarded as essential infrastructure for RollApps. The embedded AMM is the sole applicative logic on the settlement layer which is not restricted for RollApp usage only.

1 Introduction

Blockchain technology has significantly evolved since its inception. Bitcoin was the first pioneer, enabling users to create a pseudo-anonymous account for transferring balances on a distributed ledger. Ethereum further revolutionized the space by allowing turing-complete applications to be deployed on the blockchain. Cosmos was the first to focus on creating a network of networks, allowing them to interoperate with each other while retaining their own characteristics and security properties.

Despite the technological breakthroughs, these chains suffered from the subpar performance in latency and throughput of executing transactions on-chain, the overhead of bootstrapping a network and the dilutive economic effects of incentivizing security with node participation. All of those amplified the benefits of the Web 2.0 model.

The traditional web model is composed of entities paying for segregated pieces of server. For example, when Twitter servers are under heavy use it does not affect Reddit servers. Similarly, dYmension applications receive a segregated high performant execution environment running the application logic. dYmension's protocol scales by the segregation of resource consumption similar to traditional web infrastructure. However, it also allows for applications to plug themselves into an inter-connected, decentralized and trust-minimized environment. dYmension is infrastructure that glues traditional and reliable web2-like instances with the ownership and connectivity advantages of the decentralized internet.

Modular Tech

dYmension follows a modular blockchain architecture design that decouples different functions of a '*Monolithic*' blockchain for greater performance, efficiency and scalability. Monolithic blockchains handle transaction execution, state settlement, data processing and provide a consensus on the canonical history of the chain.

In dYmension each application is its own execution environment and the dYmension hub is the settlement layer. Various data availability providers will co-exist in a modular

future; thus, dYmension is designed to be data availability agnostic, enabling RollApps to easily choose data availability providers according to security and costs considerations.

In dYmension, layer specialization decouples as following:

- **Execution layer:** Processes transactions
- **Settlement layer:** Dispute resolution, maintain state roots and bridge between environments
- **Data availability layer:** Ensures all data is made available once it's published

2 Execution Layer

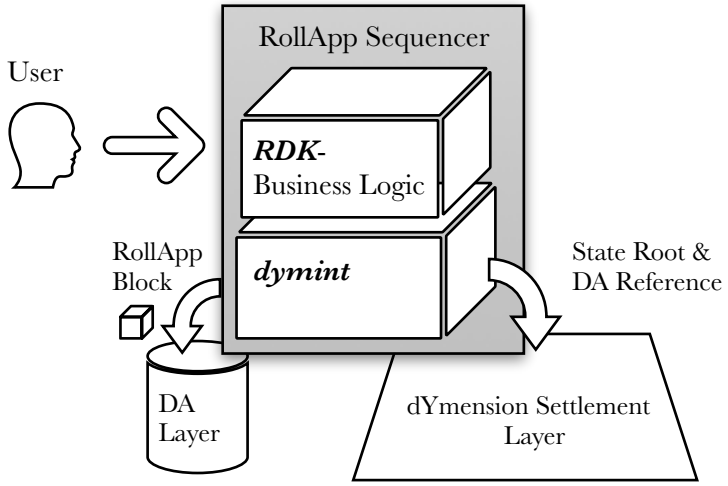
An application deployed on dYmension is an autonomous execution environment which is referred to as a RollApp. RollApps possess their own autonomy, which means they retain the benefits of building their own blockchain such as having their own native network fee token, fully flexible application logic and trust-minimized inter-operability with other RollApps and IBC-enabled blockchains. Notwithstanding, autonomy is not interchangeable with sovereignty. The divergence between the two is rooted in the security origin. Sovereign chains rely on and subsidize their own security, whereas autonomous instances natively inherit it.

dYmension RollApps exist as their own sharded execution environment in which Sequencers process transactions off-chain. A RollApp may be operated by a single Sequencer or a group of Sequencers that take turns in transaction computation. As the name implies, RollApps are derived from rollups which are designed to reduce latency and computational costs. Rollups process transactions off-chain and post transaction data and updated state roots on-chain. Data and state root publication guarantees that any actor, whether driven by distrust or by economic incentives, may independently verify that genuine computations and honest state transitions were performed by the RollApp Sequencer [1].

- **2.1.0 RollApp Architecture**

  In parallel to the modularity of the entire protocol, dYmension RollApps are disaggregated themselves. RollApps are composed out of two core services, client and server. The server is the application side designated for the RollApp deployer to implement custom business logic alongside the pre-packaged modules that construct the RollApp Development Kit. The client component, referred to as *dymint*, is responsible for block production, peer message propagation and inter-layer communication. As there are no consensus tasks in the process, dymint can provide the low latency requirements necessary for modern-day applications. A RollApp transaction lifecycle begins by sending a request to a RollApp Sequencer which processes the transaction, batches it into a RollApp block and publishes the batch and state roots to the data and settlement layers, respectively.

- **2.1.1 RollApp System Design**



- **2.2 Fraud Proof Design**

  RollApps operate under an optimistic fraud proof design, which assumes an optimistic view towards the honesty of a RollApp Sequencer, hence the term *optimistic*. However, in order to produce a trust-minimized environment, a dispute period is installed. During this dispute period other actors are able to verify that the Sequencer has submitted valid state updates. In case of a fraudulent state update, a fraud proof can be published by any permission-less actor demonstrating the valid state transition. In case the RollApp's state is proved to be fraudulent by the settlement layer, the state is reverted to the previously agreed upon state, while the bond of the malicious Sequencer is slashed and partially rewarded to the fraud proof publisher [2].

- **2.3 Sequencer Decentralization**

  dYmension introduces a system in which an application may be processed by various Sequencers, we refer to this as permissioned and permission-less PoS sequencing. As RollApps are dependent on the consensus of the settlement layer, the leader election mechanism for RollApp sequencing is derived from the relative amount of staked *DYM* (dYmension's native asset) on the settlement layer. dYmension enables RollApp deployers and governance to decide on matters such as Sequencer whitelisting criteria and other requirements that they see fit.

3 Settlement Layer

dYmension's settlement layer is built from the ground up with the mission of supporting RollApps with the utmost security, liquidity and seamless inter-connectivity. The settlement layer maintains a registrar of deployed RollApps and corresponding vital information such as state, list of sequencers, current active sequencer, execution module checksum and more, enabling the settlement layer to provide RollApps with first class service.

3.1 Enshrinement

The attribute of natively servicing rollups on the settlement layer is known as *'enshrined rollups'* [3], as the settlement layer meticulously incorporates the logic for maintaining RollApps within the layer.

Contrary to dYmension's RollApp design, non-enshrined rollups such as Arbitrum and Optimism are connected to the Ethereum base layer via a customized smart contract implementation [4]. These smart contracts are essentially multi-sig bridges which act as the interface surface between the base layer and the rollup. Buggy smart contracts and compromised multi-sig keys are only a few of the trust assumptions required for such a system. Enshrinement offers rollups the same trust and security assumptions as the base layer, yet with a simpler, safer and more efficient design space.

3.2 IRC

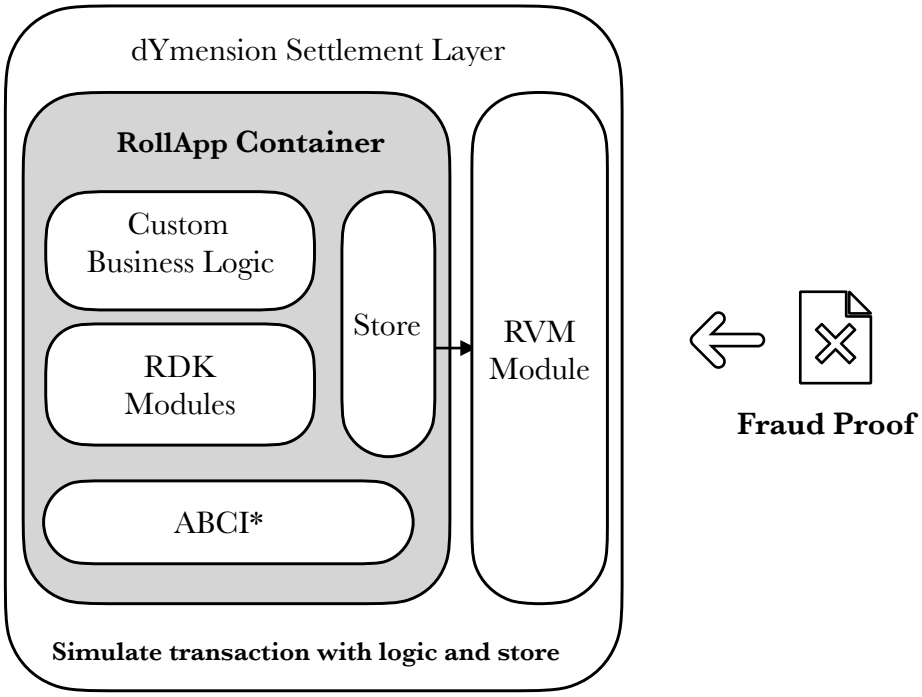
dYmension RollApps communicate and transact with each other via embedded modules which exist both on the RDK and the settlement layer. We refer to this type of communication as Inter-RollApp Communication (IRC). IRC leverages the common ground of all dYmension RollApps, which is the shared settlement layer. The settlement layer can be viewed as a hub which facilitates trust-minimized bridging for RollApp users within the ecosystem. Additionally, RollApps are capable of communicating with other IBC-enabled chains via the Hub. In effect, IRC is an internal ecosystem implementation of the IBC protocol [5] adjusted for the unique characteristics RollApps require as compared to standard Cosmos blockchains.

3.3.0 RollApp Virtual Machine

In order to service multiple rollup implementations, dYmension settlement layer spins up a virtual machine containing the RollApp execution environment upon fraud dispute. dYmension introduces a novel method for resolving disputes which is referred to as 'RollApp Virtual Machine' (RVM). RVM emulates the exact context in which a single disputed transaction was executed on a RollApp, resulting in a deterministic output. Thus, RVM is capable of resolving disputes in various execution environments.

RVM expands the capabilities and flexibility of the RollApp execution spectrum. For example, dYmension is able to service a RollApp which supports EVM execution without permanently adding EVM logic into the settlement layer. Post RVM emulation the container is discarded. RVM assures that dYmension RollApps may always evolve, experiment and keep up to date with state of the art advancements in blockchain technology and execution in particular.

3.3.1 RVM High Level Design



3.4 Censorship Resistance

dYmension's architecture is designed to mitigate the risks of censorship. Users that experience Sequencer censorship may publish a special transaction to dYmension's settlement layer. This transaction is relayed to the Sequencer with the request that it be executed within the designated timeframe. dYmension's settlement layer natively checks for the inclusion of a forced transaction via the updated state roots. If the transaction is not processed within the designated timeframe the Sequencer gets penalized.

3.5 Embedded Hub AMM

dYmension introduces an embedded Automated Market Maker (AMM) on the settlement hub, creating a core financial center. The embedded functionality is designed to expose RollApps to efficient asset routing, better price discovery and most importantly shared liquidity for the entire ecosystem. The AMM enables embedded price oracles for RollApps, which empower DeFi products based on safely priced liquidity pools. dYmension's AMM is an indispensable part of the infrastructure stack, as it provides RollApps with a critical service which can determine their usage and success. As witnessed by the role of Osmosis [6] in the Cosmos eco-system, liquidity has proven to dictate hub characteristics and has significant importance. The role of dYmension as a hub manifests with the synergy of shared liquidity on top of shared security.

4 Data Availability Layer

The dYmension protocol pursues robustness through efficient partitioning of responsibilities. The modular design focuses on off-loading data bottlenecks to deliver a highly scalable and resource segregated system. The DA layer enables any actor to verify that computation of published data has produced the same state transition specified by the Sequencer [7]. dYmension RollApps have a common interface for integrating various Data Availability (DA) providers according to security and costs preferences.

In dYmension's protocol a checkpoint is defined as a block batch published to the DA layer and the corresponding state roots to the settlement layer. Having data available and incorruptible post-checkpoint is vital as corrupted data can produce a state discrepancy between RollApp nodes. We note that reliance on a specific DA solution is highly consequential and should be perceived as a significant consideration when deploying a RollApp.

5 dYmension Economics

dYmension implements a shared security model that strengthens as more applications are deployed. A special transaction is executed on the dYmension settlement hub in order to facilitate the creation of a new RollApp with the affiliated Sequencer. Joining a RollApp as a Sequencer requires a stake of DYM tokens and a delegation to a validator of the dYmension settlement layer.

The delegation of tokens to a validator node on dYmension's settlement layer increases the portion of staked tokens and overall security of the network. Unlike standard delegation procedures Sequencers are not rewarded for delegating DYM but instead are allocated with RollApp token network fees for transaction processing.

5.1 Sequencer Stake

In order to participate as a Sequencer a minimum amount of staked DYM tokens is required. The protocol sets the minimum as an equilibrium between preventing attempts at fraud and keeping the barrier-of-entry relatively low for developers. The EV (Expected Value) of attempting fraud in an optimistic design is bound to be negative as no amount of funds can guarantee a successful attack [8]. As such, the amount of slashable stake required for bonding by the Sequencer only needs to be high enough to prevent *attempts* at fraud.

The leader election in a decentralized sequencer setup, which is a settlement layer mechanism determining the active Sequencer, is based on the relative amount of DYM tokens staked by the participants. This mechanism increases the demand for DYM as Sequencers compete over their position in the rotation. Permission-less access for RollApp sequencing creates an organic market for actors that are able to achieve the

greatest amount of profit for their assigned period. In effect, RollApps with permissionless Sequencers accrue a portion of the MEV to DYM, as a direct correlation exists between sequencing time and the relative amount of staked DYM.

5.2 RollApp Income

RollApp users interact directly with the RollApp itself resulting in user/network fees being paid according to the specific RollApp application logic. RollApp autonomy dictates network fees to be paid in the community and developer's token of choice, which can be any external token such as stable coins, DYM tokens, the RollApp's native token or any other.

RollApp logic may query external expenses and apply heuristics to hedge against cost variability. We assume that RollApp Sequencers cannot operate at a loss. As such, demand to be a Sequencer is expected to become dependent on the obtainable realizable value [9], which is derived from the following variables:

- **User fee** = RollApp gas or other collectable fee mechanics
- **Operator revenue** = User fees + MEV
- **Operator cost** = settlement layer fee + DA fee +
  Off-chain operator costs
- **Obtainable realizable value** = Operator revenue - Operator cost

As more RollApps are deployed, a flywheel effect accelerates due to the growing demand for DYM required to facilitate the following:

- AMM RollApp token swaps
- IRC and IBC transactions
- State root updates
- Growing staking requirements for Sequencers

5.3 RollApp Example

Currently, developers deploying a smart contract have a limited design space as fees are paid to the network they are transacting in. For example, every time a user submits a transaction on Uniswap a fee is paid in ETH to Ethereum. In dYmension's protocol end-users pay a fee in UNI tokens for using the Uniswap protocol. This is autonomy.

The blockchain community has seen a proliferation of tokens used for governance rather than being used for fees or real value accrual. Uniswap consistently ranks top five in ETH fee burned. However, the token holders do not benefit from the growth of the exchange's ecosystem.

As of now, Uniswap accrues zero protocol fees and if it were to change, the payment would be paid by liquidity providers. That is, traders would continue to pay a 0.30% fee on all trades and 0.05% of the amount traded would go to a "feeTo" address. Collecting this 0.05% fee at the time of the trade would impose an additional gas cost on every trade [10]. To avoid this, accumulated fees are collected only when liquidity is deposited or withdrawn. As evidenced by the Uniswap team, they are required to create workarounds to operate within Ethereum's ecosystem.

This design space limitation affects many protocols and is an impediment to progress towards sustainable protocols. dYmension expands the application design space, allowing builders to push the creativity of the blockchain space.

References

[1] An Incomplete Guide to Rollups.

https://vitalik.ca/general/2021/01/05/rollup.html

[2] (Almost) Everything you need to know about Optimistic Rollup.

https://research.paradigm.xyz/rollups

[3] Transaction Optimistic Rollups – a stepping stone for Tezos

https://research-development.nomadic-labs.com/toru-introduction.html

[4] How does Optimism's Rollup really work?

https://research.paradigm.xyz/optimism

[5] Inter-Blockchain Communication Protocol

https://ibcprotocol.org/

[6] What is Osmosis?

https://docs.osmosis.zone/overview/

[7] What is Data Availability?

https://coinmarketcap.com/alexandria/article/what-is-data-availability

[8] Optimistic Bridges

https://blog.connext.network/optimistic-bridges-fb800dc7b0e0

[9] Understanding rollup economics from first principles

https://barnabe.substack.com/p/understanding-rollup-economics-from

[10] Uniswap Whitepaper

https://uniswap.org/whitepaper.pdf