Intention Is All You Need

Kasey White
byword-hitters.9h@icloud.com
Cardano Foundation

Abstract

This paper introduces a novel approach to account abstraction on UTxO ledger based blockchains that addresses significant user and developer experience challenges in complex DeFi applications. We present a system where users sign intentions—fragments of desired transaction elements—rather than complete transactions. Unlike traditional UTxO systems where addresses are derived directly from keys, our smart contracts store key information in on-chain state, enabling a quorum of keys to control an address while validating that user intentions are fulfilled in the final transaction. This approach provides increased security through multi-signature capabilities, simplifies user interaction with blockchain assets, and enables new DeFi applications that were previously impractical on UTxO systems, such as efficient orderbook exchanges, atomic swaps, and sequencer services for interactions with global state.

Keywords Cardano · UTxO · Account Abstraction · Blockchain

1 Introduction

Blockchain users often tie airdrops, protocol rewards, stake control, asset ownership, and user identity to a single address. In traditional UTxO-based blockchains like Cardano and Bitcoin, these addresses are typically formed from a single key, creating potential for significant security vulnerabilities. If that key is lost or stolen, all associated funds are compromised. Additionally, interacting with malicious sites can lead to complete fund drainage through deceptive transaction signing requests.

This paper introduces Bullet, the first comprehensive approach to account abstraction on UTxO-based blockchains that fundamentally transforms how users interact with their assets. Rather than managing individual UTxOs and signing complete transactions, users sign intention messages that specify only fragments of what they want included in a transaction and the fee they're willing to pay.

Our technical contributions include:

- A novel intention validation scheme specifically designed for Cardano's execution model
- Multi-signature support with customizable quorum requirements using hot and cold keys
- An account-centered approach that simplifies both wallet interfaces and DApp interactions
- Fund shielding through an innovative vault mechanism that protects assets from DeFi vulnerabilities

Bullet has been fully implemented and is publicly available on GitHub. Our work combines the security benefits of UTxO-based systems with the usability advantages typically associated with account-based blockchains, while introducing new capabilities through the intention validation mechanism.

2 Background

Account abstraction fundamentally aims to separate blockchain address ownership from transaction execution logic. This separation enhances user experience by providing flexibility in how transactions are authorized and executed, while maintaining security. In account-based blockchains like Ethereum, account abstraction has evolved through various standards (such as ERC-4337), enabling features like social recovery, sponsored transactions, and scheduled payments.

2.1 UTxO Model Challenges for User Experience

Cardano, Bitcoin, and other UTxO-based blockchains face unique challenges in implementing account abstraction due to their fundamental transaction model. In the UTxO (Unspent Transaction Output) model, blockchain state is represented as a collection of discrete outputs, each with a unique identifier. Transactions consume existing UTxOs as inputs and create new UTxOs as outputs.

This model creates specific challenges that account abstraction could address:

1. **Fragmented Value Storage**: Users must manage multiple UTxOs rather than a single account balance, complicating wallet interfaces and user experience.
2. **Global State Contention**: When multiple users attempt to interact with a UTxO representing global state (such as a DEX liquidity pool), only one can succeed while others' transactions fail, requiring users to sign entirely new transactions.
3. **Complex Transaction Construction**: Applications requiring interaction with multiple UTxOs demand precise transaction construction with specific inputs and outputs, placing significant burden on wallet implementations.
4. **Single-Key Security Risks**: Traditional UTxO addresses derived from single keys create security vulnerabilities where key compromise leads to total fund loss.

Despite these challenges, the UTxO model offers important security advantages: transaction outcomes are explicitly defined within the transaction itself rather than computed during execution. This provides deterministic results and makes certain types of attacks (like reentrancy) structurally impossible since all inputs and outputs are specified before validation.

2.2 Intentions as a Solution Approach

To address these challenges while preserving UTxO security benefits, we introduce the concept of intentions. An intention is a signed message containing a user's authorization to perform a specific action within a transaction, without specifying the complete transaction structure.

The basic structure of an intention includes:

- The specific action to be performed
- The fee the user is willing to pay

- The address responsible for paying the fee
- The signature(s) validating the intention

This approach allows users to authorize specific actions without constructing entire transactions. The technical implementation details of how intentions are represented, validated, and integrated into the transaction flow will be covered in subsequent sections.

By shifting from transaction signing to intention signing, we create a foundation for account abstraction in UTxO blockchains, enabling improved user experiences while maintaining the security properties that make UTxO models valuable.

3 Bullet

Bullet is an account abstraction system for UTxO-based blockchains implemented as a series of micro-validators. This architecture distributes functionality across six main actions—hot spend, cold spend, vault spend, intention validation, credential changes, and account deletion—each implemented as a separate validator to minimize execution costs on Cardano. These validators are linked via a proxy contract that checks for the execution of the appropriate specialized contract based on the requested action.

The system enables users to interact with their assets through intention signatures rather than complete transaction signing, addressing the contention and UX issues in UTxO-based blockchains while maintaining their security properties.

3.1 Hot and Cold Multisig Credentials

Bullet implements key management through hot and cold multisignature credentials:

Hot Keys provide flexibility for day-to-day operations:
- Normal spend transactions (e.g., sending assets to friends)
- Signing intentions for DeFi interactions
- Actions related to staking or voting in blockchain governance
- Limited access to Vault UTxOs (requiring a higher quorum threshold of signatures)

Cold Keys offer enhanced security for critical operations:
- Access to spending when hot keys are unavailable (e.g., offline servers) or lost
- Authority to modify credential state, including updating both hot and cold key sets

The quorum threshold for hot keys and wallet operations is configurable by the user based on their security needs and can be modified through the change credentials action. For cold keys, the quorum is always equal to the total number of cold keys, ensuring maximum security for critical operations.

The system supports multiple signature schemes including Schnorr, ECDSA, and Ed25519, providing compatibility across various wallet implementations and key management practices.

3.1.1 Signing a Transaction

When signing a transaction in Bullet, users follow a process similar to traditional UTxO transaction signing, but with support for multiple signatures. The transaction hash is generated and then signed by the required quorum of keys according to the transaction type (hot spend, vault spend, or cold spend). This multisignature capability enhances security while maintaining a familiar workflow for users.

For recovery scenarios where a user loses access to some hot keys, the cold keys serve as a backup mechanism, allowing users to update their credential state with new hot keys through the change credentials action.

3.2 Fund Management

Bullet's Vault mechanism allows users to designate specific UTxOs that require elevated security for spending:

- Vault UTxOs require either a greater quorum of hot key signatures or cold key authorization
- This creates a separation between freely available funds and secured assets
- Users can maintain a small portion of funds for normal spending or DeFi interactions while protecting the majority of their holdings
- The mechanism helps protect against draining attacks or misuse by DApps

This structure enables off-chain indexers to track how much value Bullet users allocate to DeFi applications versus secured storage, potentially serving as an indicator of DeFi ecosystem health.

For a user to specify a UTxO as a Vault UTxO, they simply need to create that UTxO with a Vault type datum from the Bullet Datum types.

3.3 Intentions in Bullet

Intentions are the core innovation of Bullet, changing how transactions are authorized in UTxO-based systems. Unlike traditional transactions where users sign complete input-output structures, an intention is a signed message that specifies constraints the final transaction must satisfy.

Bullet validate these constraints through a validation engine with these components:

Constraint System:
- Output constraints: specifying address, value, datum, and reference requirements
- Input constraints: validating existing UTxOs being consumed
- Signature requirements: ensuring proper authorization
- Minting policies: controlling token creation
- Time bounds: enforcing valid time ranges
- Contract Execution: enforcing the presence of contract executions in the transaction

Validation Flow:
- Analysis of redeemer data containing intention lists
- Per-user intention processing including signature validation
- Constraint system validation against transaction structure
- Value movement tracking and verification
- Final verification of intention completion

Execution Models:
- Sequential execution: using incrementing nonce values
- Parallel processing: using UTXO inputs as nonces
- Combined models supporting multiple users with different execution patterns

The constraints are limited to simple equality checks (with the exception of time bounds) and don't support general inequality predicates like less than or greater than. To include more DeFi use cases, a temporary value system was added and is detailed below.

3.3.1 Temporary Value System

The temporary value system passes data between constraints during intention validation:

- Starts as `None` and can be set by certain constraints (input, reference input, redeemer validation)
- Subsequent constraints can use this value for validation decisions or output construction
- Once used, each constraint typically updates the temp_val for the next constraint

This system allows intentions to reference dynamic values that aren't known at signing time, such as cryptocurrency prices from an oracle or wallet balances. For example, a swap intention could use a price obtained from an oracle contract execution rather than requiring a hardcoded value at signing time. This enables intentions to adapt to on-chain conditions without requiring users to sign new intentions.

3.3.2 Signing an intention

An intention in Bullet consists of specific data elements:

1. A list of constraints (output requirements, input validation rules, etc.)
2. The maximum fee the user is opting into
3. A nonce to prevent replayability of signed intentions

The signing process works as follows:

1. These data elements are serialized into a byte array with appropriate wrapping for compatibility
2. The byte array is hashed
3. The resulting hash is signed by the required quorum of keys
4. The final transaction includes all original intention data, the signatures, and the user's address

At transaction execution, the validator compares each constraint in the intention against the actual transaction structure, rejecting the transaction if any constraint is not met. This validation step prevents any attempt to use signed intentions for operations that exceed their authorized scope.

3.3.3 Cost Model Maybe (Stretch Goal)?

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut.

4 Applications in DeFi

Bullet's intention-based approach solves key challenges in DeFi applications on UTxO-based blockchains, enabling more efficient and user-friendly experiences across multiple use cases.

4.1 Orderbook DEX

Traditional orderbook DEXs on UTxO blockchains face significant limitations:

- Users pay transaction fees for placing orders upfront, even if they're never executed
- Order contention requires signing new transactions when targeted orders are already taken
- Market responsiveness is limited by blockchain confirmation times

Bullet addresses these issues through its intention mechanism:

- Users create orders via intentions and only pay when orders execute ("pay-on-execution")
- Multiple orders can be created rapidly to follow market movements
- A single signed intention remains valid regardless of UTxO contention
- The fee is satisfied from any available UTxOs, eliminating the need for re-signing when transactions fail

This approach creates a more efficient orderbook experience by removing upfront costs for unexecuted orders, reducing overall transaction fees, and improving market responsiveness for traders.

4.2 Atomic Swap

Bullet enhances atomic swap capabilities by allowing multiple intentions to be combined in a single transaction with flexible user conditions. Key benefits include:

- Multiple parties can participate in complex exchanges without requiring direct coordination
- All-or-nothing execution ensures transactions either complete fully or fail completely
- New use cases like atomic flash loans become possible, where borrowed value must be returned within the same transaction
- Similar to other Bullet operations, users only pay fees when the transaction is successfully included on the blockchain, not when loan attempts fail
- Risk is minimized as failure results in no value movement or fee payment

This creates more robust peer-to-peer trading possibilities while maintaining the security guarantees of atomic execution.

5 Interoperability

Bullet supports cross-chain key compatibility through multiple signature schemes to create a more accessible user experience.

To reduce friction for users with existing blockchain wallets, Bullet allows for Secp256k1 and Schnorr signature schemes alongside the native Ed25519 keys used in Cardano. This means users can interact with Cardano using the same private keys they already use for Bitcoin, Ethereum, and many EVM chains, eliminating the need to generate and manage separate keys for each blockchain ecosystem.

The implementation uses an alternative message signing approach necessitated by the constraints of the UTxO model. Since providing external signatures to a transaction would alter the transaction hash itself, Bullet derives the message to be signed from the script context available during on-chain execution:

1. The on-chain Plutus script accesses its execution context
2. Inputs and outputs from this context are serialized
3. This serialized data is hashed to create a consistent message
4. The resulting message can be signed by any supported key type
5. The signature can be provided to the transaction without changing the message hash

This approach represents a tradeoff: it enables cross-chain key compatibility but provides less control than traditional transaction signing since users sign only the inputs, outputs, and fees rather than the complete transaction hash including metadata.

While broader protocol-level interoperability features are not currently implemented and specific wallet integrations are not yet in development, the key compatibility layer provides an important foundation for future cross-chain user experiences. When implemented in wallets, this would allow users to access and manage Cardano assets using the same wallet interface and keys they already use for other blockchains.

5.1 Sequencer Ordering

When accessing global state in UTxO blockchains, contention creates significant user experience challenges. Bullet transforms this model:

- Users sign intentions to utilize global state without directly handling contention
- Transaction builders (sequencers) combine multiple user intentions into single transactions
- Sequencers, better equipped for rapid signing, handle contention for global state UTxOs
- Users experience reliable interaction regardless of underlying contention

Sequencers are incentivized through fees included in the intentions themselves. These fees can be denominated in any token, allowing for flexible payment models. The cost for including an intention can be calculated in advance, creating predictable economics for both users and sequencers.

This approach is particularly valuable for high-demand protocols where global state access (like liquidity pools or lending markets) creates bottlenecks in traditional UTxO designs.

6 Conclusion

This paper presented Bullet, a new approach to account abstraction for UTxO-based blockchains. By replacing transaction signing with intention signing, we've created a system that combines UTxO security with account-model convenience.

Our approach separates what users want to accomplish from how transactions are constructed. This separation addresses several fundamental challenges in UTxO blockchains:

- It eliminates the need to manage discrete UTxOs
- It resolves contention issues when accessing global state
- It enables flexible multi-key security policies

The applications we've outlined demonstrate real solutions to existing problems rather than theoretical improvements. Pay-on-execution for orderbooks, efficient atomic swaps, and contention-free access to global state all derive directly from the intention-based architecture.

Integration with DexHunter is in progress, with plans for Lace integration soon. These implementations will provide practical validation of the approach.

Future work will focus on quantum resistance through script-executed quantum-resistant signature verification schemes.