CARRY: ADVANCING THE FUTURE OF GAMES V1.0

ANDY PAN[1]

Feb, 2024

CONTENTS

LIST OF FIGURES

[1] *CTO of Carry, Singapore*

ABSTRACT

Carry introduces an innovative protocol tailored for the game industry, offering the potential for increased revenue for game developers and players. By replacing traditional game platforms, Carry protocol provides powerful features for building, conducting analysis in games, bridging Web2 and Web3 games, and placing in game advertisement.

The Carry Protocol introduces a transformative approach to integrating advertising into the world of web3 games. Through the innovative concept of slots - virtual spaces, soundscapes, or digital entities within games - it offers an unintrusive avenue for presenting advertisements. These slots, inherently tied to specific game environments, can manifest as visual displays, auditory cues, or even interactive virtual entities. Given their intrinsic value, slots are not just static elements but tradable assets within the Carry ecosystem. The paper delves deep into the mechanics of slots, elucidating their lifecycle, potential monetization strategies through auctions, and the importance of performance metrics in optimizing their impact. Moreover, the role of the Carry token and Carry fuel underpins the economic dynamics of the system, ensuring seamless transactions within the Carry marketplace. This protocol harnesses the power of blockchain and decentralized governance, promoting a balanced, sustainable, and player-centric advertising ecosystem in the game world.

Carry Protocol also offers a robust platform equipped with an Software Development Kit (SDK) and powerful modular tools, enabling game developers to efficiently manage and monetize digital assets such as fungible and non-fungible tokens (FTs and NFTs). The Carry SDK enables Web2 games to integrate blockchain technology easily. At the same time, Carry SDK provides a simplified mechanism for managing in-game assets and user accounts for both Web2 and Web3 games.

1 INTRODUCTION

Carry Protocol aims at building a next generation game infrastructure. The infrastructure aims to fill the gap between Web2 and Web3 games, at the same time creating a fair game economics for blockchain games to thrive as never before. Briefly, Carry introduces an innovative advertising protocol tailored for the game industry, offering the potential for increased revenue for game developers and players. By replacing traditional advertising platforms, this protocol not only enhances the advertising effectiveness for advertisers but also reduces associated costs.

The Carry Protocol innovatively bridges the worlds of games and advertising within the web3 space, offering a unique system for showcasing ads within virtual game environments. These advertising spaces, or "slots," are thoughtfully integrated into games, functioning like digital billboards for interactive promotions. This approach aims to create a better and healthier game advertising economics.

Moreover, the protocol incorporates a transparent and equitable auction system to allocate advertising space, where the market value of each slot is determined by real-time demand. This system ensures a fair and open process for all participants.

The Carry SDK on the side, will provide powerful infrastructure for both Web2 and Web3 games to easily integrate blockchain features. Furthermore, Carry provides adaptive data analysis module for various kinds of games to achieve strategy optimization and service evolution.

1.1 Motivation

In this section, we attempt to summarize the primary challenges of the current game market.

- **Failed Incentives:** The dominance of the play-to-earn dual-token economic model in many web3 games, although initially promising, has demonstrated its limitations. Due to its aggressive and singular incentive structure, it often places developers and players in positions of discordant interests, threatening the game's longevity [1]. In the later stages of the vast majority of play-to-earn games, due to the singularity of incentive mechanisms, players can solely profit by selling their held NFT assets. Regardless of the developers' efforts to salvage or enhance the game experience, games remain highly susceptible to entering a "death spiral" within a relatively short timeframe.

  For a sustainable future, web3 games necessitate diversified economic mechanisms that foster collaborative wins among developers, players, and other stakeholders.

- **Platform Monopoly:** Developers have grown weary of the "Apple tax", which, as a giant advertisement platform, dominates the web2 advertising system and web2 monetization standards by monopolizing user data assets [2]. The web2 advertising ecosystem primarily comprises four key players: advertisers, media, advertising platforms, and users. Advertising platforms, by monopolizing user data assets, shape the advertising monetization landscape. However, the advancement of privacy technology and blockchain technology has provided a mature technological solution for a fairer advertising protocol [3].

Both developers and users have a strong demand for a more equitable and efficient advertising ecosystem. The current dominance of giant advertising platforms has significantly impacted business and innovation efficiency.

- **Inability to Analyze Player Intent:** Despite the transformative potential of web3 in games, many developers operate based on speculative insights about player preferences, leading to games that either wane in popularity or serve primarily for asset arbitrage. The heart of this challenge is the absence of effective tools to analyze user behavior within Web3 games. This oversight hinders developers from truly understanding player intent [4]. Notably, games like Crypto Kitties, which introduced users to the creation and trade of in-game NFTs, underscore the vast possibilities but also the need for more nuanced player behavior insights.

  Using the crypto-native game 'Loot' as a case study, it's evident that following its significant traction within the NFT community, over 20 development teams have emerged with a focus on Loot. These teams are diligently working on an array of games rooted in the foundational model of Loot. However, a minimal number of these endeavors have shown tangible advancement. To solve these issues, the developers and players need to have more interaction during its lifecycle. Native features in web3, for example, predict the market, provide methods for developers to truly interact with players.

  As a slogan in the web3 industry goes, "onchain is the new online." Therefore, a multitude of data dimensions can now be easily captured. We need greater insights into users to enhance the efficiency of advertising and games.

1.2 Related Works & Projects

The Carry Protocol operates at the intersection of blockchain technology, gaming, and digital advertising, marking a significant step forward in the evolution of Web3 gaming infrastructures. To contextualize Carry's contributions, it's essential to review existing projects and research that have paved the way for innovations in game finance (GameFi), in-game advertising, and blockchain integration within the gaming industry. This section delves into several key projects and areas of work that share thematic or technical overlap with the aims of Carry Protocol.

1.2.1 Decentraland and Virtual Real Estate

Decentraland represents a pioneering effort in creating a decentralized virtual world, where users can buy, sell, and manage virtual real estate [5]. It stands as a hallmark example of integrating blockchain technology with digital land ownership, offering a vivid parallel to Carry's slot-based advertising model. In Decentraland, landowners have complete control over their parcels, which can be developed to display content, host games, or serve advertisements, similar to Carry's vision of in-game slots as digital real estate for ads.

1.2.2 Axie Infinity and Play-to-Earn Model Evolution

Axie Infinity revolutionized the GameFi space by popularizing the play-to-earn (P2E) model, where players earn cryptocurrency rewards for gameplay achievements and trading in-game assets [6]. While Carry Protocol acknowledges the limitations of the early P2E models, it seeks to refine this concept by creating a more sustainable and equitable economic system

for all stakeholders, emphasizing the need for diversified economic mechanisms beyond the dual-token models that Axie initially introduced.

1.2.3 Brave Browser and Attention-Based Advertising

The Brave Browser and its associated Basic Attention Token (BAT) introduce a novel approach to online advertising, where users are rewarded for viewing ads [7]. This model aligns closely with Carry's ethos of fair compensation for engagement. Carry extends this philosophy into the gaming domain, proposing a system where players and developers benefit directly from ad integration, suggesting a move towards more interactive and player-focused advertising strategies within the game industry. Moreover, Carry provides a general infrastructure for both traditional and emerging (Web3) games to integrate blockchain features, which is not achieved by the Brave Browser.

1.2.4 Enjin and Asset Tokenization

Enjin has been a frontrunner in providing an ecosystem for creating, managing, and integrating tokenized digital assets into games and apps. By allowing assets to be tokenized on the blockchain, Enjin facilitates true ownership of in-game items, mirroring Carry's concept of slots as tradeable digital assets [8]. This parallel underscores the broader trend towards asset tokenization in gaming, enhancing the depth and liquidity of virtual economies.

1.2.5 The Sandbox and User-Generated Content

The Sandbox is a user-generated content (UGC) and gaming platform that empowers players to create, own, and monetize their gaming experiences on the Ethereum blockchain [9]. It exemplifies the potential of blockchain to democratize content creation within virtual worlds. Carry Protocol's slot-based advertising model can be seen as an extension of this democratization, where advertising spaces become a canvas for creative expression and economic participation by developers and players alike.

1.2.6 Aavegotchi and DAO Governance in Gaming

Aavegotchi stands out for its integration of decentralized finance (DeFi) principles with non-fungible tokens (NFTs) within a gaming context, governed by a Decentralized Autonomous Organization (DAO) [10]. The project highlights the potential for community-driven governance models in shaping game development and economic policies, resonating with Carry's aim to foster collaborative wins among all participants in the gaming ecosystem.

1.2.7 Comparison Analysis

In examining these related works and projects, it's clear that the Carry Protocol is not operating in isolation but rather building upon a rich foundation of innovation in blockchain, gaming, and digital advertising. By learning from the successes and limitations of these precedents, Carry aims to forge a new path that addresses the challenges of failed incentives, platform monopolies, and the need for deeper insights into player behavior. As the blockchain and gaming landscapes continue to evolve, projects like Carry Protocol will play a crucial role in shaping the future of digital entertainment, advertising, and community engagement.

1.3 Contributions

1.3.1 Slot-based Advertising Paradigm

At the heart of the Carry Protocol lies the innovative concept of a "slot," a transformative idea designed to bridge the worlds of advertising and on-chain games. In its essence, a slot is a unique digital real estate where advertisements can be strategically placed, integrated seamlessly into the virtual environments of games or other digital platforms.

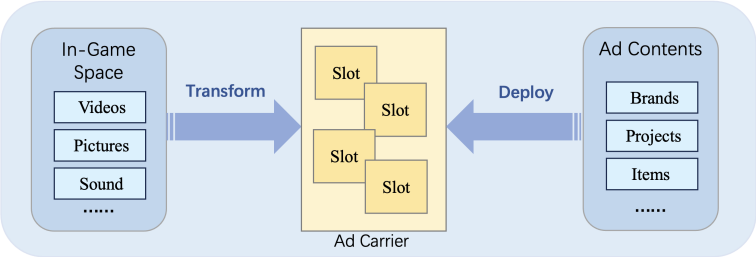

Figure 1: Slot Design

- Every slot is inherently tied to a designated game, taking forms such as visual spaces for graphics, character-specific sound effects, or virtual companions roaming the digital environment.
- Beyond their in-game presence, slots hold asset value and can be traded or transferred among participants.
- Slots can be a carrier of various types of media, videos, pictures etc.

Imagine you're playing an online game, and as you explore, you see ads on billboards or even on items like a character's backpack. These are what we call "slots" in the Carry Protocol. Think of a slot like a piece of virtual real estate where ads can live. These slots are valuable to advertisers and can be bought, sold, or traded just like other virtual items in the game.

Slots are versatile. They can show video ads, become part of the game scenery, or even be part of the outfit your character is wearing. They're designed to fit into the game naturally, so they're interesting rather than annoying.

The worth of these ad spaces can go up or down. For example, a slot right where players hang out the most could be worth a lot because more players will see the ad there, while a slot off in a quiet corner might not be as valuable.

The Carry Protocol is all about changing up the game when it comes to ads. These slots are active parts of the game that can change based on how players interact with them and what advertisers want to try. They're a new way to think about ads in games, making sure that players' experiences come first.

1.3.2 One-Stop Game Infrastructure

As early blockchain experts and game developers merged, there appeared to be innumerable hours spent learning seemingly nonessential concepts foreign to each other's native

skillsets. Many developers suggested that fortes were being unnecessarily overcomplicated. They needed a tool to realize their function; comprehending its mechanism should be optional education.

Without a proper foundation, an ecosystem cannot thrive. Carry SDK was developed precisely for this purpose; it's a secure and easy-to-use library of standardized blockchain tools, or rather, a one-stop metaverse infrastructure platform for the game ecosystem. Using Carry SDK, developers no longer need to be intimidated or hindered by the security and complexity of blockchain technology. The entry threshold is lowered, and in terms of cost, eliminated. All interested gamers can expect increased playability as a result.

1.3.3 Adaptive Game Data Analysis

The Carry Protocol introduces an Adaptive Data Analysis Module for Web3 gaming, offering real-time insights into in-game advertising effectiveness through Performance Benchmark Metrics. This innovation enables game developers and advertisers to optimize ad placements and content dynamically, ensuring ads enhance rather than detract from the gaming experience.

By leveraging strategy simulation and comprehensive evaluation metrics, the protocol facilitates precise adjustments and targeted experiments with advertising strategies. This approach not only improves monetization strategies for developers but also ensures advertisements are engaging and relevant to the player community, marking a significant advancement in the integration of blockchain technology with the gaming industry.

1.4 Organization

The reminder of this Whitepaper are organized as follows. First, we introduce the proposed novel slot-based advertising paradigm, the general launching infrastructure for both Web2 and Web3 games, and the adaptive data analysis module in Section 2-4, respectively. Second, we demonstrate the designed economics and incentives for various system participants in Section 5. Third, we describe the contract-based protocol implementation and some cryptography-facilitated mechanisms in Section 6. Finally, we conclude this work in Section 7.

2 PROPOSED NOVEL ADVERTISING PARADIGM

2.1 Overview

The Carry Protocol introduces something called "slots" in online games, which are like digital billboards or sponsored items within the game. These slots add value without getting in the way of gameplay and allow advertisers to get more bang for their buck.

Normally in games, the game makers control everything you see and use. But in these new games, players can own unique game items, like a special sword or a piece of land, which they can buy, sell, or trade. These items are called NFTs, which are like digital collectibles that you truly own.

Game developers must establish clear ownership guidelines when introducing slots as in-game advertising spaces. There are two categories of slots: those that are inherent to the game's environment (game native assets) and those that are controlled by players (player owned assets). For instance, a billboard within the game's landscape serves as a game native asset. Contrastingly, player-specific items such as character skins or pets, which are created and consequently 'minted' by players, represent player owned assets.

To elucidate, let's consider a hypothetical scenario. A digital realm, 'LandA', owned by 'Addr1', becomes a canvas for an advertisement. When the developer introduces a slot to this realm, it isn't just assigned to the player 'Addr1'. Instead, it's as if 'LandA' itself becomes enhanced, now comprising both the original realm and the new slot. Symbolically:

$$\text{Addr1} \rightarrow \text{Land A}$$

Transforms to:

$$\text{Addr1} \rightarrow (\text{LandA}, \text{SlotA})$$

This distinction is crucial. The slot isn't an isolated entity but rather correlates with a specific in-game asset. It isn't a generic add-on but a tailored integration. And to facilitate this tailored relationship, technologies like EIP-6220 can be leveraged, ensuring that the connection between the game asset and its corresponding slot is both seamless and efficient.

In summary, Carry Protocol is re-imagining how ads fit into video games. Instead of being annoying interruptions, they're becoming a natural part of the game's world. This setup lets game creators and players work together in new ways, making the game more engaging while also figuring out who gets to own these digital ad spaces. It's a fresh approach that could change games and advertising for the better.

2.2 Basic Definitions

Within the Carry Protocol, several foundational concepts drive its operation in the web3 games world. This section elucidates these essential principles, highlighting their role and importance in the Carry ecosystem.

1. **Slot:** The fundamental unit within Carry, serving as the primary canvas for advertisement placements.
2. **Slot Time:** A designated time frame for which a slot can be utilized, regulated by the Carry governance.

3. **Auctioning:** The dynamic process by which slots' time and positioning are bid upon, determining their allocation based on market demand and value perception.
4. **Marketplace:** A decentralized platform where slots can be traded, purchased, or leased, fostering a vibrant ecosystem around the advertisement spaces.

2.3 Concrete Designs

The central tenet of the Carry Protocol hinges on the intricate lifecycle of slots. These slots are not mere placeholders but dynamic entities that embody a unique relationship with game assets and advertisements. Here's a closer look at their lifecycle:

- **Creation and Initialization:** Game creators are given special slots that they can easily put into their games. Once these slots are in place, the creators or the players can start using them, turning them into one-of-a-kind digital items.
- **Asset Status Management:** The protocol oversees and modifies the status of each asset, ensuring up-to-date representation within the ecosystem.
- **Asset Relationship Management:** The system smartly manages how different parts of the game work together, particularly when something changes. For example, when a slot stops showing an ad, the ad is no longer connected to that slot.

2.3.1 Slot Status

- **Idle:** The slot is in a passive state, devoid of any advertisements.
- **Placement:** The slot is actively displaying an advertisement.
- **Pre-initialization:** The slot awaits proper initialization or minting, making it unowned within the ecosystem.
- **Initialization:** Pertains to the formal creation of a slot for a specific game asset, represented technically by the minting of the slot NFT.

2.3.2 Slot Relationships

Two primary relationships define a slot's existence:

- **Ownership Dynamics:** Dictates the proprietorship of the slot, tracing it to either a developer, gamer, or directly linking it to a specific in-game asset. At the outset, this ownership typically aligns with a distinct game asset.
- **Advertisement Affiliation:** Emphasizes the bond between the slot and the advertisements it hosts. The intrinsic value of a slot is not just its mere existence, but its capacity to exhibit advertisements and, in turn, generate revenue.

2.3.3 Slot-Ad Lifecycle

Examining slots from an advertising temporal perspective helps understand how they accrue value in the ad ecosystem:

- **Slot Placement Duration Auction:** Taking cues from platforms like OpenSea, slots undergo an auction or selection process. Unlike traditional auctions where ownership is transferred, slots are typically leased for a predetermined duration, conferring advertisement display rights without altering slot ownership.
- **Ad Content Placement:** Once a slot is chosen, it's imbued with the pertinent advertising content.
- **Value Generation:** The advertisement, once viewed or engaged with by users or gamers, starts accruing value.
- **End of Ad Cycle:** Upon reaching the designated period, the advertisement's active phase ceases, reverting the slot to its idle state.
- **Ad Effect Tracking:** This phase is earmarked for gauging the repercussions and reach of the advertisement.
- **Ad Settlement:** Beyond the initial fee amassed during the auction, any supplementary value spawned by the advertisement's efficacy is settled at this juncture.

2.3.4 Actions on Slots

Several actions can be taken on a slot, determining its trajectory and interaction with ads:

- **Ad Placement:** Assigning a specific advertisement to the slot for display.
- **Ad Removal:** Extracting the currently showcased advertisement from the slot, returning it to a vacant state.
- **Transfer of Rights:** Conveying the privileges or ownership of the slot to another entity.
- **Slot Exchange:** Engaging in transactions to exchange slots with other participants.

2.3.5 Placement Strategy

The strategy of placement ads onto slots determines how slots accrue value and relevance within the game ecosystem. As developers and advertisers navigate this space, optimizing placement strategies ensures slots' maximum potential is realized.

Carry provides the following built in strategies:

- **Distinct Time Allocation:** Slots are auctioned for exclusive, well-defined time periods.
- **Periodic Time Allocation:** Slots are systematically scheduled for recurring durations.
- **Enduring Lease:** The slot is granted on a long-term basis to a specific party. Nonetheless, the lessee has the prerogative to auction this lease in the Carry marketplace.

2.4 Governance Mechanism

Imagine Carry Governance as the manager of a digital billboard. It matches ads with the right spots in the virtual game world, ensuring that everything fits together nicely and the game experience stays enjoyable and cohesive. Carry governance includes the following functions:

- **Slot Allocation Governance:** Determines and assigns specific virtual spaces for advertisement displays.
- **Content-Slot Matching:** Regulates and ensures the relevant advertisement content is mapped to its corresponding slot.
- **Ad Lifecycle Management:** Governs the duration and lifecycle of an advertisement within a slot.
- **Ad Quality Oversight:** Establishes standards and ensures displayed advertisements meet quality and relevance criteria.
- **User Feedback Integration:** Incorporates feedback mechanisms and uses player responses to refine ad placements.
- **Dispute Resolution:** Manages and resolves conflicts that may arise in terms of slot assignments or advertisement displays.

2.5 Ad Settlement Mechanism

Beyond the foundational fee secured during the auction phase, any additional value derived from the ad's performance is settled. The settlement mechanism can be divided into:

1. **Basic Placement Fee (**BPF**)**:
$$Fee = BPF \times t \tag{1}$$
where t is the actual duration of the ad placement.
2. **Performance-based Incentives (**PM**)**: Depending on the agreement, this could be:
   - **CPS (Cost Per Sale)**: For on-chain businesses, e.g., NFT sales.
   - **CPA (Cost Per Acquisition)**: Based on user metrics.
   - **CPL (Cost Per Liquidity)**: In scenarios like DeFi, based on staked asset value.
   - **CPI (Cost Per Installation)**: Metrics based on app installations.

Ultimately, the revenue (R) for a slot owner is:

$$R = A \times (BPF \times t) + B \times \Sigma PM \times \Sigma PR \tag{2}$$

where PR represents the Performance Ratios, and A and B are coefficients. Slot owners can adjust these coefficients to maximize their returns.

3 DATA ANALYSIS MODULE FOR GAMES

Blockchain games combine elements of traditional gaming with features of decentralized finance to create a new gaming and economic experience. In this context, on-chain data analysis becomes particularly important as it can offer insights into player behavior, asset liquidity, economic activities, and the health of the gaming ecosystem. Data analysis can bring several key advantages to GameFi projects:

- **Transparency and Verifiability:** Since all transactions and activities are recorded on the blockchain, data analysis can provide fully transparent and verifiable insights into player behavior and economic activities.
- **Player Behavior Insights:** Analyzing on-chain data can reveal player preferences, engagement patterns, and behavioral trends, helping developers optimize game design and enhance player experience.
- **Economic and Asset Analysis:** Monitoring and analyzing the creation, trading, and liquidity of in-game assets allows developers to better manage the game's economy, preventing inflation or asset value collapse.
- **Community and Ecosystem Health:** On-chain data analysis helps assess community activity and engagement, identify key participants and contributors within the ecosystem, thus fostering community growth and sustained engagement.

For Carry Protocol, offering data analysis functions in addition to its slot-based advertising features would enable it to provide deeper market insights to game developers and advertisers, optimize advertising strategies, and promote the healthy development of the entire gaming ecosystem. In this way, Carry Protocol can enhance the effectiveness of its advertising platform while supporting the growth and success of GameFi projects through data-driven insights. The framework of Carry data analysis module is illustrated in Figure 2.

3.1 Analysis Objectives

The evolution of GameFi necessitates a data-driven approach to improve user experience, maintain economic stability, and optimize advertising strategies. Carry Protocol addresses these needs through its robust analytics. The primary objectives can be summarized as follows.

- **Enhancing User Experience:** Analyzing in-game behavior and transaction patterns helps developers understand the user journey. Data reveals which game features captivate players or where they face obstacles. This insight enables developers to iterate on game design, ensuring players find joy and challenge in equal measure, fostering long-term engagement.
- **Economic Stability and Inflation Prevention:** Carry Protocol's analytics aim to track the velocity of in-game currencies and the balance between sinks and sources. Monitoring these economic indicators provides the foresight needed to make adjustments that mitigate inflation, keeping the in-game economy vibrant and fair.
- **Optimizing Ad Slot Performance:** Advertisement slots are a crucial revenue source in GameFi. Data analysis on slot performance, including user engagement and ad reach,

informs strategies to place and design ads that resonate with players without disrupting their gaming experience, ensuring ad slots contribute positively to the game's revenue without detracting from its playability.

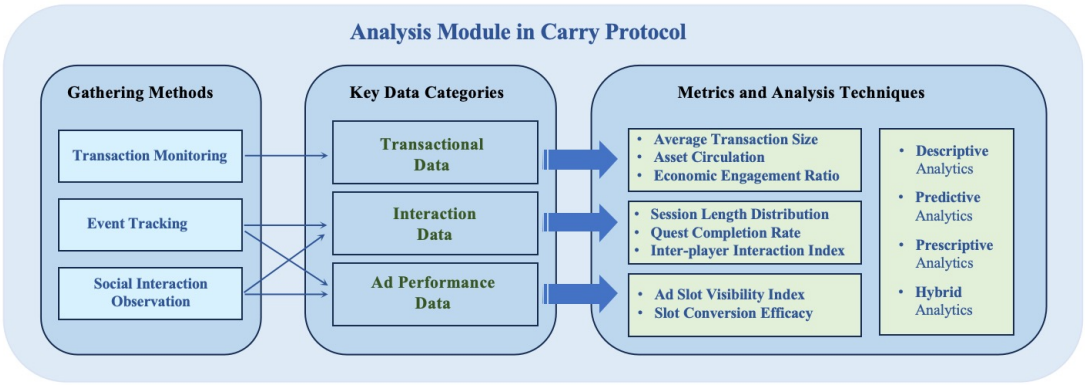

Figure 2: Data Analysis Framework in Carry Protocol

3.2 Data Types and Metrics

In this section, we provide detailed data types and significant performance metrics according to the analysis objectives mentioned above.

3.2.1 Key Data Categories

For each analytical objective, specific data types are necessary to provide actionable insights.

1. **Transactional Data:** Every in-game transaction, from item purchases to currency exchanges, reflects the game's economic health. Analyzing this data helps in identifying trends, detecting anomalies, and understanding the flow of virtual economies.
2. **Interaction Data:** Player interactions, both with the game environment and with other players, shed light on community dynamics and engagement levels. This data is invaluable for community management and for enhancing multiplayer aspects of the game.
3. **Ad Performance Data:** Data on ad views, clicks, and conversions is essential for measuring the effectiveness of in-game advertising. This data category helps advertisers understand the impact of their ads and adjust campaigns for maximum engagement.

3.2.2 Performance Benchmark Metrics

Performance Benchmark Metrics within the Carry Protocol are carefully crafted indicators that provide actionable insights derived from specific on-chain data categories. These metrics are designed to evaluate the effectiveness of both the in-game economy and advertising strategies. Here are several refined metrics derived from three aforementioned data types:

- **From Transactional Data:**

1. Average Transaction Size: Indicates the average amount of currency used in transactions, reflecting player spending habits and economic health.
2. Asset Circulation: Measures how frequently in-game assets change hands, revealing the liquidity and dynamism of the game's market.
3. Economic Engagement Ratio: Compares active players to economic transactions, identifying how deeply players are engaged with the in-game economy.

- **From Interaction Data:**
  1. Session Length Distribution: Shows the range and average lengths of time players spend in-game per session, indicating engagement depth.
  2. Quest Completion Rate: Tracks the percentage of completed in-game quests, measuring content engagement and potential areas for content optimization.
  3. Inter-player Interaction Index: Quantifies the interactions between players, such as trades or cooperative play, highlighting the game's social dynamics.
- **From Ad Performance Data:**
  1. Player Response Time: Records the time it takes for players to interact with an ad after it appears, indicating the initial impact and relevance of the ad content.
  2. Slot Conversion Efficacy: Analyzes the rate at which ad impressions lead to desired player actions, such as in-game purchases or sign-ups, offering a direct measure of ad effectiveness.

The resulting metrics serve distinct purposes. For Game Developers, they enable fine-tuning of game mechanics to improve player retention, balance the in-game economy, and enhance overall player satisfaction. For Advertisers, these metrics provide a granular understanding of how different player segments interact with advertisements, informing targeted marketing strategies and creative ad content development. Ultimately, these metrics not only illuminate the current state of game and ad performance but also inform predictions and guide future improvements. They are the linchpin for optimizing GameFi experiences and ensuring that both players and developers reap the maximum benefits from their interactions within the ecosystem.

3.3 Data Collection and Analysis Techniques

3.3.1 Gathering Methods

The granularity and accuracy of data collection determine the quality of insights derived.

- **Event Tracking:** This method is leveraged for player interaction data, including player actions, in-game achievements, quest completions, etc. Capturing in-game events provides a detailed picture of player behavior and game mechanics efficacy, allowing for granular analysis and immediate feedback for iterative development. Through an SDK or API integrated into the game client, player operations and game events are captured in real-time. Then, each event is tagged and categorized for deep analysis, such as player progress, preferences, and areas of friction.

- **Transaction Monitoring:** This method is leveraged for economic activity data, encompassing player-to-player transactions, purchases of virtual items, currency liquidity, etc. By real-time recording and indexing of all in-game and inter-game transactions using smart contracts and blockchain event logs, Carry Protocol can provide real-time data on economic activities, facilitating a responsive approach to economic management within the game. Finally, it enables developers to monitor and adjust in-game economic policies, prevent inflation, and ensure the health and sustainability of the game economy. Figure 3 introduced by wipro represents a blockchain monitoring framework.

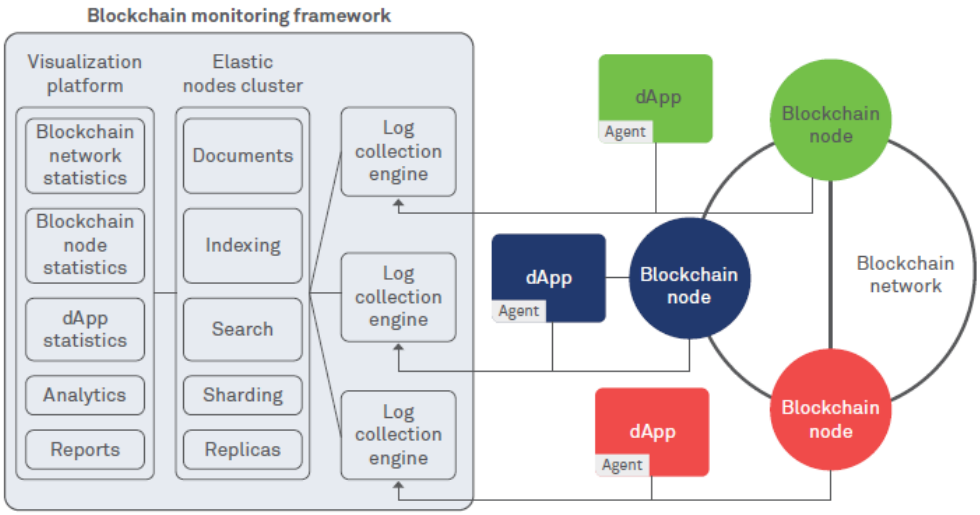

Figure 3: Transaction Monitoring Framework

- **Social Interaction Observation:** This method is leveraged for community engagement data, such as communications between players, collaboration, and formation of social networks within the game. These data collected by analyzing players' in-game chat logs, teamwork, and social interaction events. Insights into the community's vibrancy, social structure, and player engagement levels can be gleaned from social interaction data. Developers can design more attractive social features and events based on these insights, encouraging interaction and collaboration among players.

3.3.2 Analytical Methodologies

Each data type requires a specific analysis approach to transform raw data into meaningful insights.

- **Descriptive Analytics:** This foundational method summarizes raw data into interpretable formats, establishing a baseline understanding of in-game and economic activities. For example, we can aggregating transactional data to compute average transaction size, volume, and frequency over specific time intervals. By analyzing these aggregates, developers can understand the peak times of economic activity within the game, identify spending patterns, and potentially discover underutilized areas of the game economy that could be developed further.

- **Predictive Analytics:** Using statistical models and machine learning algorithms, predictive analytics forecasts future trends, informing proactive game and economic development strategies. For example, we can employ machine learning models, such as decision trees or neural networks, to predict player churn based on interaction data like session length, frequency of play, and engagement in community events. These predictions enable developers to proactively implement features or incentives aimed at retaining players at higher risk of churn, effectively increasing overall engagement and player lifetime value. For example, Google Cloud proposed to use BigQuery ML on the sample app dataset to predict propensity to user churn or not churn based on users' demographics and activities within the first 24 hours of app installation.

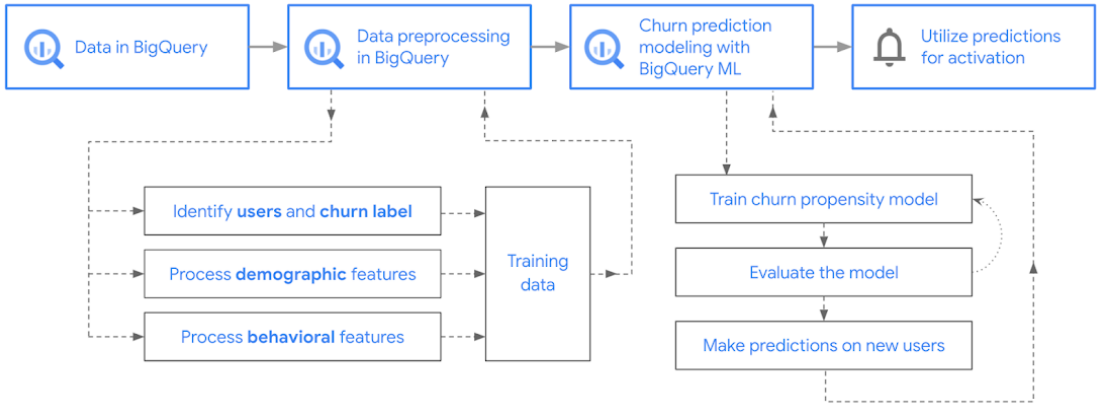

Figure 4: Predictive Analytics Framework Proposed by Google Cloud

- **Prescriptive Analytics:** Beyond predicting future trends, prescriptive analytics suggests concrete actions to achieve desired outcomes, such as improving user retention or optimizing ad placements. For example, we can utilize advanced analytics, such as optimization algorithms or simulation models, to analyze click-through rates (CTR), conversion rates (CR), and ad exposure duration across different player segments. Insight: This analysis can suggest which ad content is most effective for specific player demographics or in-game contexts, guiding advertisers on how to tailor their campaigns. For example, if certain ad placements consistently lead to high conversion rates among players who enjoy PvP content, similar strategies can be applied to target this player segment more effectively.

- **Combining Methodologies for Comprehensive Insights:** By integrating descriptive, predictive, and prescriptive analytics, Carry Protocol can offer a holistic view of the game's ecosystem. For instance, descriptive analysis of transactional data may reveal an increasing trend in virtual item purchases. Predictive analytics could then forecast this trend's continuation based on current game dynamics and player behavior. Finally, prescriptive analytics could recommend specific in-game events or promotions to capitalize on this trend, such as limited-time offers on popular items or introducing new items likely to be popular based on player preferences inferred from past data.

By addressing these objectives, categories, and methodologies, Carry Protocol's data analysis SDK will empower developers and advertisers with the tools needed to unlock the full potential of GameFi. This structured approach ensures that every facet of the gaming experience and economy can be analyzed, optimized, and enhanced for the benefit of all stakeholders in the ecosystem.

4 GENERAL INFRASTRUCTURE FOR BOTH WEB2.0 AND WEB3.0 GAMES

At the most basic level, Carry SDK assists developers in integrating blockchain technology into both traditional and emerging Web3 games. It does not mean building every blockchain game from scratch to completion, which takes a massive amount of time and is a non-standardized process that serves only a limited number of users. From the perspective of efficiency, we want this integration to be a standardized and minimized process.

Specifically, the proposed infrastructure is composed of three fundamental modules (Asset Management Module, Identity Management Module, and Security Protection Module) and two novel modules (Ad Managment Module and Data Analysis Module).

4.1 Asset Management Module

The Asset Management Module aims to revolutionize the ownership and tradeability of in-game assets through blockchain technology. By leveraging fungible tokens (FTs) for in-game currencies and non-fungible tokens (NFTs) for unique items, this module ensures true ownership, provenance, and interoperability of assets across different gaming platforms and ecosystems. The core designs are composed of the following three elements:

- **Tokenization:** Utilizing ERC-20 (for FTs) and ERC-721 or ERC-1155 (for NFTs) standards to represent in-game currencies and items on the blockchain.
- **Smart Contracts:** Deploying smart contracts to handle the logic for asset creation, ownership transfer, and transactions, ensuring transparency and security.
- **Blockchain Layer:** Integration with a blockchain layer (e.g., Ethereum, Binance Smart Chain) for decentralized asset management and to leverage its security protocols.

Based on the elements mentioned above, the processes can be summarized as follows:

1. Define asset classes and attributes in smart contracts.
2. Deploy contracts to the blockchain, generating a unique address for each asset.
3. Implement game logic to interact with blockchain for asset transactions using Carry SDK.
4. Use event listeners for real-time updates on asset state changes within the game environment.

4.2 Identity Management Module

This module seeks to establish a decentralized identity system that allows game players to use a single, secure, and persistent identity across both Web2 and Web3 gaming platforms. By incorporating Decentralized Identifiers (DID) [11], it aims to enhance user privacy, security, and cross-platform interoperability. The core designs are composed of the following three elements:

- **DID Integration:** Utilizing the DID standard for creating verifiable, self-sovereign digital identities tied to blockchain addresses.

- **Wallet Binding:** Facilitating the binding of a player's wallet to their game account, enabling seamless in-game and blockchain interactions.
- **SDK Features:** Providing APIs within Carry SDK for easy integration of DIDs, supporting login, authentication, and asset management.

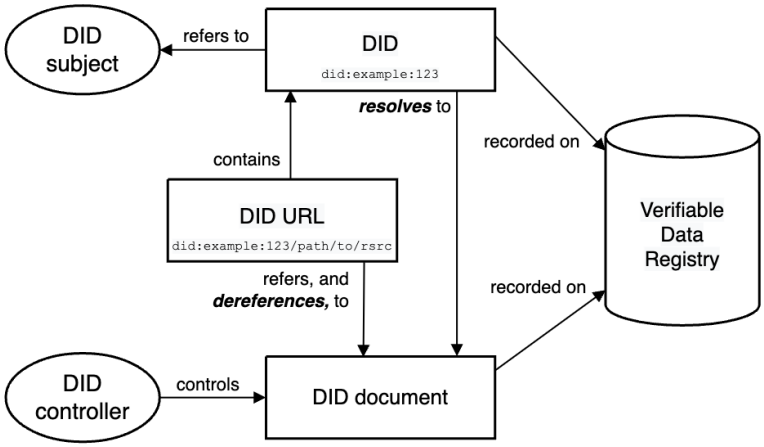

Figure 5: Overview of DID architecture and the relationship of the basic components.

We introduce a basic overview of the major components of Decentralized Identifier architecture provided by W3C, as shown in Figure 5. Based on the elements mentioned above, the processes can be summarized as follows:

1. Initialize DID for each user upon registration using Carry SDK.
2. Bind user's wallet address to their game account, linking in-game and blockchain identities.
3. Authenticate user actions in-game and on blockchain via digital signatures.
4. Store and manage user's digital assets and identity securely on-chain.

4.3 Security Protection Module

The focus is on ensuring the integrity and security of in-game transactions and digital asset management. By incorporating advanced cryptographic techniques such as multi-party computation (MPC) [12] and zero-knowledge proofs (ZKP) [13], this module aims to protect user assets, maintain privacy, and secure transactions without compromising on user experience. The core designs are composed of the following three elements:

- **Cryptography:** Implementing MPC for secure, distributed private key management and ZKP for transaction validation without revealing sensitive information.
- **Smart Contract Security:** Employing security practices like audits and formal verification to ensure smart contract integrity.

- **SDK Security:** Ensuring the Carry SDK incorporates the latest security protocols for interaction with blockchain networks, including secure API calls and encryption of sensitive data.

Based on the elements mentioned above, the processes can be summarized as follows:

1. Implement MPC protocols for key management, allowing transactions without exposing private keys.
2. Use ZKP for transaction validation, ensuring privacy and security.
3. Conduct regular security audits and update smart contracts and SDK accordingly.
4. Integrate secure, encrypted communication channels within Carry SDK for asset and identity management.

4.4 Technical Architecture for Carry SDK

The above modules collectively provide a robust framework for integrating blockchain technology into gaming, offering a seamless bridge between traditional (Web2) and blockchain-based (Web3) gaming experiences. Through these advancements, Carry Protocol aims to enhance ownership, interoperability, and security for the gaming community. In this section, we demonstrate the detailed technical architecture for Carry SDK, as shown in Figure 6.

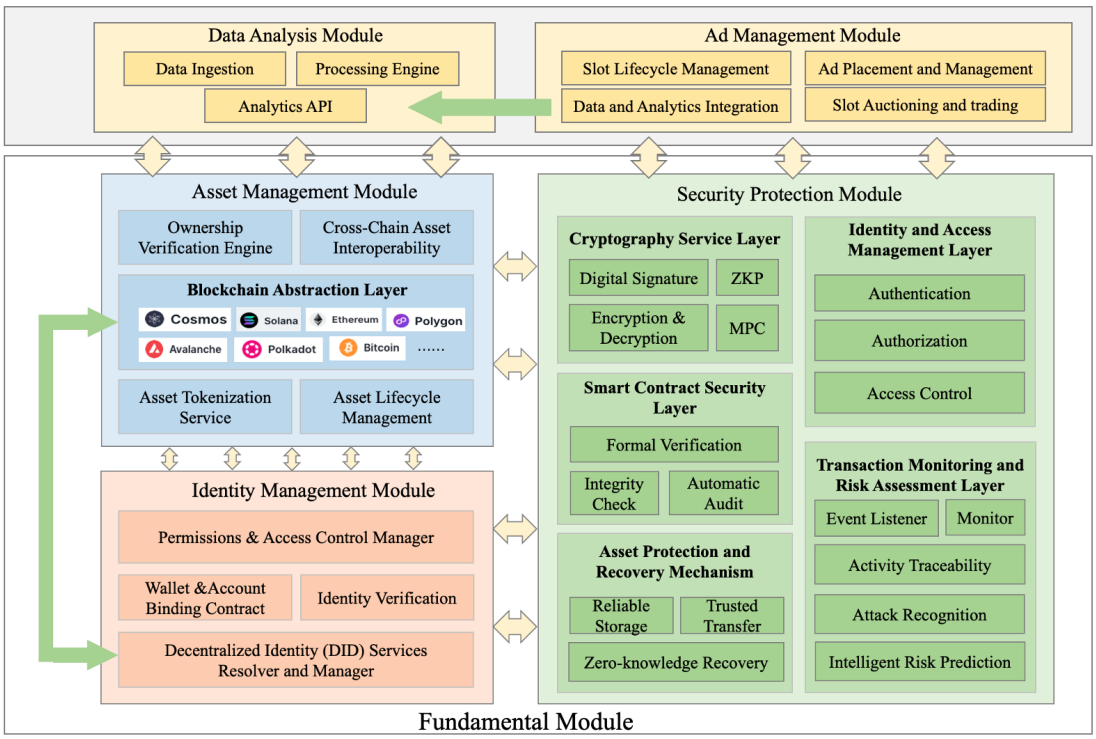

Figure 6: Technical Architecture Overview of Carry SDK Modules.

4.4.1 Asset Management Module

There are five key components in Asset Management Module. We explain them in a bottom-up manner.

- **Asset Tokenization Service (ATS):** Converts in-game assets into blockchain tokens. It supports multiple standards (ERC-20, ERC-721, ERC-1155) and manages the lifecycle of tokens including creation, transfer, and destruction.
- **Blockchain Abstraction Layer (BAL):** Serves as the interface between game servers and various blockchain networks, abstracting the complexity of blockchain protocols and offering a unified API for asset transactions. Interfaces with various blockchains to abstract complexities, providing a unified API for asset transactions across networks like Cosmos, Ethereum, Solona, Polygon, and Binance Smart Chain.
- **Ownership Verification Engine (OVE):** OVE verifies the ownership and authenticity of in-game assets through immutable transactions. This verification process is critical for preventing fraud, duplication, or unauthorized access to assets, thereby ensuring that only rightful owners can initiate transfers or modifications. By providing a trustless mechanism for ownership verification, OVE plays a key role in maintaining a secure and fair gaming environment, where players have full confidence in the value and authenticity of their digital possessions.
- **Cross-Chain Asset Interoperability (CCAI):** Cross-Chain Asset Interoperability (CCAI) addresses one of the key challenges in the blockchain space: enabling assets to move freely and securely across different blockchain networks. CCAI facilitates the trading and exchange of assets between disparate blockchains, enhancing their liquidity and usability. This interoperability is crucial for creating a unified gaming economy where assets from one game or platform can be utilized or traded in another, regardless of the underlying blockchain. By breaking down the barriers between blockchain ecosystems, CCAI enables a more interconnected and fluid digital asset market, where players can leverage the full potential of their in-game assets across the blockchain universe.

4.4.2 Identity Management Module

There are five key components in Identity Management Module.

- **Decentralized Identity Services:** Decentralized Identity Services form the backbone of the identity management module in Carry Protocol SDK, focusing on managing user identities through decentralized identifiers (DIDs). These services are responsible for the creation, resolution, updating, and revocation of DIDs. By associating a user's DID directly with their game account, it facilitates cross-platform identity verification and usage. DIDs serve as a universal identity marker across different games and platforms, enabling a seamless gaming experience. This system ensures that identities are portable, self-sovereign, and can be verified without reliance on centralized authorities.
- **Wallet and Account Binding:** The Wallet and Account Binding component is tasked with securely linking a user's blockchain wallet address to their gaming account. This process is integral for enabling in-game identity verification and asset management through DIDs. By binding the wallet address with the user's DID, it ensures that the user can

leverage their blockchain identity within the gaming environment. This linkage is crucial for authenticating transactions and interactions within the game, ensuring that in-game assets and achievements are securely tied to the user's blockchain identity, facilitating a transparent and trustless ecosystem for asset ownership and exchange.

- **Identity Verification:** Identity Verification is a critical process within the identity management module that ensures the authenticity of users by leveraging their DIDs. This component utilizes cryptographic methods to verify that actions, transactions, or access requests are genuinely initiated by the rightful owner of the DID. The process involves validating the user's digital signatures and ensuring that the DID associated with a game account matches the blockchain-verified identity. This layer of verification is pivotal for maintaining the integrity of in-game interactions, preventing impersonation, and ensuring that all transactions are securely authenticated.
- **Permissions & Access Control Manager:** The Permissions & Access Control Manager governs the access rights of users within the gaming environment, based on their DIDs and predefined access policies. This component is essential for defining and enforcing the rules that determine what resources a user can access and what actions they can perform within a game. By leveraging the user's DID for access control, it ensures that only verified and authorized users can interact with sensitive in-game assets or perform certain operations. This mechanism enhances the security of in-game assets and data, preventing unauthorized access and ensuring that game environments remain safe and fair for all participants.

4.4.3 Security Protection Module

There are five primary components in Security Protection Module.

- **Cryptography Service Layer (CSL):** The Cryptography Service Layer (CSL) is a foundational component of the security guarantee module, dedicated to providing robust encryption and decryption services. It supports advanced cryptographic technologies, including digital signatures for verifying the integrity and origin of data, as well as Zero-Knowledge Proofs (ZKP) to facilitate transactions and validations without compromising user privacy. By leveraging ZKP and other cryptographic methods, CSL plays a crucial role in ensuring that sensitive user information and transaction details remain confidential, thereby bolstering privacy protection across the blockchain network.
- **Smart Contract Security Framework (SCSF):** The Smart Contract Security Framework (SCSF) focuses on the development and deployment of secure smart contracts, incorporating automated security audits and formal verification processes to identify and rectify common vulnerabilities. This framework ensures that smart contracts, which govern the logic and execution of blockchain transactions, are devoid of loopholes that could lead to unauthorized access or asset theft. By subjecting all smart contracts to rigorous testing and verification, SCSF ensures the integrity of the contracts, safeguarding assets against exploitation and enhancing overall contract security.
- **Identity and Access Management Layer (IAM):** The Identity and Access Management Layer (IAM) manages user identities and access rights, utilizing Decentralized Identity

Verification (DID) to enhance security measures. By employing DIDs alongside multi-factor authentication, IAM ensures that only authenticated users can access sensitive operations and data. This approach to identity management not only strengthens security by verifying user identities but also enhances user control over personal information and privacy, ensuring secure and authorized access within the blockchain ecosystem.

- **Transaction Monitoring and Risk Assessment Layer (TMRAS):** The Transaction Monitoring and Risk Assessment Layer (TMRAS) is tasked with the real-time surveillance of blockchain transaction activities. Utilizing machine learning and behavioral analysis techniques, TMRAS assesses the risk levels of transactions, identifying and mitigating potential threats of fraud and theft. This proactive approach to transaction security allows for the timely detection of suspicious activities, ensuring the protection of assets and maintaining the integrity of the blockchain network.
- **Asset Protection and Recovery Mechanism (APRM):** The Asset Protection and Recovery Mechanism (APRM) provides a robust defense against unauthorized access and attacks on user assets. Offering solutions for asset recovery in the event of security incidents, APRM ensures the safe storage and transfer of user assets. By implementing measures to quickly restore assets following a breach or loss, APRM plays a critical role in maintaining user trust and confidence in the blockchain's ability to secure digital assets against potential threats and vulnerabilities.

4.4.4 Ad Management Module

Following the design of core modules in the Carry SDK framework, such as the Asset Management Module, Identity Management Module, and Security Guarantee Module, Carry Protocol introduces a new component: the Ad Management Module. This module is designed to offer an innovative advertising paradigm based on "slots" in blockchain games, integrating advertisements naturally within the game while delivering value to both game developers and advertisers. There are four core components of the Module Framework.

- **Slot Lifecycle Management:** This component is responsible for overseeing the entire process of a slot's lifecycle, from creation and initialization to the display and updating of advertisements. Game developers can integrate special slots into their games with ease. Once these slots are in place, they can be utilized by developers or players, transforming them into unique digital items. Monitors and updates the status of each slot, ensuring accurate representation within the ecosystem. Intelligently manages the cooperation between different game elements, especially when changes occur.
- **Ad Placement and Management:** Controls the display of advertisement content across various slots, including the addition, removal, and refresh of ads. Once a slot is selected, relevant advertising content is imbued within it. Assesses the impact of displayed advertisements on users or gamers, beginning to generate value.
- **Slot Auctioning and Trading:** Provides a mechanism for the time and positioning of slots to be bid upon and allocated based on market demand and perceived value. Following operations similar to platforms like OpenSea, slots undergo an auction or selection process. Slots are typically leased for a specified duration rather than transferring ownership.

- **Data and Analytics Integration:** Closely integrates with the Data Analysis Module, offering in-depth analysis of ad effectiveness and slot performance. Utilizing the functionalities of the Data Analysis Module, metrics such as click-through rates and conversion rates are analyzed to provide precise advertising strategies for advertisers.

By incorporating the Ad Management Module as part of the Carry SDK framework, the Carry Protocol not only redefines the way advertisements are incorporated into video games but also provides new avenues for collaboration between game creators and players. This innovative approach promises to make the gaming experience more engaging while addressing the ownership of digital ad spaces, potentially bringing positive transformations to both the gaming and advertising industries.

4.4.5 Data Analysis Module

In addition to the above components, the Data Analysis SDK is structured to seamlessly work within the Carry Protocol ecosystem, enhancing its functionality with dedicated components for comprehensive data analytics. This module is composed of the following three primary components:

- **Data Ingestion Module:** This component is responsible for capturing both on-chain data and in-game activity data. It employs interfaces that facilitate the streaming of diverse data types into the system, ensuring that data is accurately and efficiently collected for analysis.
- **Processing Engine:** At the core of the SDK is the Processing Engine, which utilizes advanced algorithms for real-time data analysis. This engine processes data from various sources, applying descriptive, predictive, and prescriptive analytics to transform raw data into actionable insights.
- **Analytics API:** The API provides endpoints for accessing the analytics results, allowing developers and advertisers to query specific data points and retrieve insights. This component ensures that the data analysis results are accessible, interpretable, and ready for application.

By integrating the Data Analysis Module within the Carry SDK framework, the Carry Protocol ensures that game developers and advertisers have access to a comprehensive suite of tools for data-driven decision-making. This holistic approach not only enhances the gaming experience and economic viability but also enriches the advertisement strategy, making the Carry Protocol a robust foundation for GameFi projects.

4.4.6 Module Interactions

The Carry SDK orchestrates seamless interactions among its five core modules: Identity Management, Asset Management, Security Protection, Ad Management, and Data Analysis. Central to this ecosystem is the Identity Management Module, which authenticates users and links their identities to digital assets and transactions, forming the basis for secure interactions within the SDK. The Asset Management Module utilizes this identity framework to tokenize in-game assets and manage their ownership, ensuring assets are securely linked to users. The Security Protection Module enhances this setup by providing cryptographic security and integrity checks, safeguarding assets and user data against potential threats. Integrating with

these foundations, the Ad Management Module leverages user and asset data to place and optimize in-game advertisements, enhancing player experience and advertising value. The Data Analysis Module complements this by analyzing data across modules, offering insights to refine user engagement, asset management, and ad effectiveness.

Together, these modules create a symbiotic environment where secure asset tokenization, personalized advertising, and comprehensive analytics converge to offer a robust and user-centric blockchain gaming experience.

5 ECONOMICS AND INCENTIVE MECHANISMS

The Carry token introduces several new features:

Governance Vote: Each Carry token represents a vote in the platform's governance process. The more tokens held in a wallet, the greater the holder's influence and voting power on proposals.

Medium of Exchange: All transactions within the platform can be settled using Carry tokens, with fees and rewards generated during the transaction process distributed in Carry tokens.

Collateral: Carry token holders can use their tokens as collateral to access additional services provided by the platform, including identity management, asset management, security protection, and SDK provision.

Membership: Participants can obtain membership through payment and by providing collateral, which includes access to data analysis, market advice, and additional services such as identity management, asset management, security protection, and SDK provision.

Auction Marketplace Creation: The ad auction process within the Carry ecosystem is vulnerable to several hard-to-prevent attacks, including collusion and witch hunts. Users can use Carry tokens as collateral to create an auction marketplace. The marketplace creator can adjust its parameters, and manage the transaction process, but cannot manipulate the transaction outcomes. The main responsibility of the creator is to regulate participant behavior and to mitigate certain attacks that are currently difficult to prevent against the VCG mechanism. Furthermore, the creator's behavior will be monitored, and abnormal activities will trigger a penalty vote.

Carry-Game Token AMM Pool: The Carry platform will offer services for numerous games. In the future, we will provide a game asset Dex service, allowing users to exchange assets across various games using Carry tokens as an intermediary. Token holders can create an AMM pool for liquidity mining by providing collateral.

5.1 Governance and Collateral

Carry aims to be a decentralized service platform for games. Virtually all interactions within the platform, including parameter designs, development direction, auction methods modifications, auction market creation, and various auction mechanism parameters, will be determined through votes by Carry token holders. The voting process is directly proportional to the number of Carry tokens held, granting token holders proportional voting rights and influence over proposals.

Users can provide Carry tokens as collateral to gain more rights. Basic collateral rights include access to additional services provided by the platform's three modules (identity management, asset management, and security protection), contributions from SDK users in the ecosystem, and penalties for misconduct. Moreover, users can collateralize their assets to earn revenue by creating and managing an auction marketplace. In the future, after establishing the game asset exchange pool, users can also provide liquidity to the exchange pool to earn mining rewards.

5.2 Customized Auction Market

The auction process in the Carry ecosystem is susceptible to several hard-to-defend attacks, including collusion and witch hunts. Users can collateralize Carry tokens to establish an auction marketplace, where the creator can adjust its parameters and oversee the transaction process without influencing the outcomes. The creator's primary role is to regulate participant behavior to mitigate certain VCG mechanism attacks that are currently indefensible. The creator's actions will be under surveillance, and any detected abnormal behavior will prompt a penalty vote. Auction marketplace creators could be individual game operators selling in-game ad space or ad space traders managing their traffic. All Carry token holders are eligible to create an auction marketplace, provided they collateralize sufficient Carry tokens and abstain from malicious practices.

5.3 Auction Mechanisms

Carry platform provides personalized advertising auction marketplace creation form, so we provide all kinds of mainstream auction mechanism services, including open incremental auction (British auction), open decremental auction (Dutch auction), GFP auctions, GSP auctions, VCG auctions, etc., and additional design of the Dutch auction mechanism and VCG auction mechanism.

5.3.1 Incremental auction (English auction)

An Incremental Auction is a type of auction where participants openly bid against each other, with bids increasing in value until no higher bids are offered. This is in contrast to sealed-bid auctions, where bids are submitted privately.

In an Incremental Auction:

1. Open Bidding: The auctioneer starts the bidding process at a certain price, often the minimum acceptable bid. Participants then openly raise the bid in increments, announcing their bids aloud or indicating them through gestures or signals.

2. Incremental Bidding: Bids must exceed the current highest bid by a predefined increment. The auctioneer may announce these increments, and participants must adhere to them when raising their bids.

3. Competitive Bidding: Bidders continue to raise the price until no one is willing to offer a higher bid. This competitive process typically results in the highest possible price for the item being auctioned.

4. Winner Determination: The participant who offers the highest bid when no further bids are made wins the auction and is obligated to pay the final bid amount.

5. Transparency and Participation: Incremental auctions offer transparency as all participants can see each other's bids in real-time, allowing them to make informed decisions about whether to continue bidding.

Incremental auctions are commonly used for selling various types of goods, including antiques, artwork, real estate, and other high-value items. They are also employed in online platforms and platforms for advertising space, where bids are placed electronically and in real-time. The open nature of incremental auctions promotes competition among bidders, often leading to higher prices compared to other auction formats.

5.3.2 Dutch Auction (Open Descending Price Auction)

In a Dutch Auction, the price of the ad slot starts high and decreases over time until an advertiser places a bid or the price drops to a reserve price.

AGENCY STRATEGY FOR DUTCH AUCTION: The agency aims to secure the ad slot for its client at the lowest possible price but also needs to balance the risk of losing the slot to another bidder. The strategy could be defined as follows:

$$B_i(t) = \begin{cases} 0 & \text{if } P(t) > V_i \\ V_i & \text{if } P(t) \leqslant V_i \text{ and } \text{Risk}(t, V_i) > \theta \\ 0 & \text{otherwise} \end{cases}$$

Here:

- $B_i(t)$ is the bid from agency i at time t
- $P(t)$ is the current price at time t
- V_i is the agency's valuation for the ad slot
- $\text{Risk}(t, V_i)$ is a function evaluating the risk of losing the slot if waiting any longer, considering the current time and valuation
- θ is a risk threshold, beyond which the agency decides to bid to avoid losing the slot

In a more sophisticated approach, the agency not only considers the current price and its own valuation but also takes into account the time decay, estimated competition, and a dynamic risk assessment.

$$B_i(t) = \begin{cases} 0 & \text{if } P(t) > V_i \\ V_i \cdot \left(1 - e^{-\lambda \cdot (T-t)}\right) & \text{if } P(t) \leqslant V_i \cdot \left(1 - e^{-\lambda \cdot (T-t)}\right) \text{ and } \text{Competition}(t, P(t)) \leqslant \gamma \\ 0 & \text{otherwise} \end{cases}$$

Here:

- T is the total time duration of the auction
- λ is a parameter controlling the time sensitivity of the bid
- $\text{Competition}(t, P(t))$ is a function estimating the level of competition based on the current time and price
- γ is a competition threshold, beyond which the agency decides not to bid due to high competition

In this strategy, the agency increases its willingness to bid as the auction progresses, adjusting its bid according to the time decay function. The agency also evaluates the level of competition at the current price and decides to bid only if the estimated competition is below a certain threshold.

RISK FUNCTION AND COMPETITION FUNCTION DESIGN To address the dynamic and complex nature of bidding in a Dutch auction, where the price of an ad slot decreases over time, we propose sophisticated models for evaluating the risk of losing the slot and estimating the level of competition, i.e., $\mathrm{Risk}(t, V_i)$ and $\mathrm{Competition}(t, P(t))$.

The risk of losing the ad slot increases as the auction progresses and as the current price approaches the agency's valuation of the slot. We model this risk as a function of time and the difference between the valuation and the current price, incorporating market competition. It is calculated as:

$$\mathrm{Risk}(t, V_i) = \frac{1}{1 + e^{-\kappa \cdot (V_i - P(t)) \cdot \omega(t)}} \tag{3}$$

where:

- κ controls the steepness of the function, reflecting how valuation differences affect risk perception.
- $\omega(t)$ is a weight function that increases with time, indicating the growing risk as the auction nears its end. The calculation formula is shown in Equation (4).

$$\omega(t) = \frac{T - t}{T} \tag{4}$$

Competition dynamically changes based on multiple factors including the current price, time elapsed, and historical bidding behavior. As the price decreases, the auction may attract more bidders, affecting the competition level. The calculation formula is defined as:

$$\begin{aligned} \mathrm{Competition}(t, P(t)) &= \beta \cdot \log\left(\frac{P(0)}{P(t)}\right) + \eta \cdot \frac{t}{T} \\ &= \beta_0 \cdot \left(1 + \mu \cdot \frac{dP}{dt}\right) \cdot \log\left(\frac{P(0)}{P(t)}\right) + \eta \cdot \frac{t}{T} \end{aligned} \tag{5}$$

where:

- $P(0)$ represents the initial price of the ad slot at the start of the auction.
- β represents the effect of price decrease on attracting more competition. It can be modeled as a function related to the rate of price decrease, enhancing our understanding of how price dynamics influence competition levels during the auction.

$$\beta = \beta_0 \cdot \left(1 + \mu \cdot \frac{dP}{dt}\right) \tag{6}$$

  where:

  - β_0 is the base competition attractiveness coefficient, indicating the level of competition without price changes.
  - μ is a modulation coefficient that reflects the sensitivity of competition attractiveness to the rate of price decrease.
  - $\frac{dP}{dt}$ represents the rate of price decrease, indicating the change in price over unit time.

- η captures the impact of time on competition, reflecting how urgency among bidders may increase as the auction progresses.

These models provide a more accurate representation of the strategic considerations in a Dutch auction, allowing agencies to make informed decisions based on the evolving risk and competition levels.

ROI OF AGENCIES' STRATEGY In a Dutch Auction, the price of the advertising slot decreases over time, and agencies aim to secure slots at an optimal price point that balances cost and potential returns. An advanced bidding strategy should therefore consider both the cost of the slot and the expected return on investment (ROI).

The bid function $B_i(t)$ for agency i at time t can be formulated as:

$$B_i(t) = \begin{cases} 0 & \text{if } P(t) > V_i \text{ or } ROI_i(t, P(t)) < ROI_{\text{threshold}} \\ P(t) & \text{if } P(t) \leqslant V_i \text{ and } ROI_i(t, P(t)) \geqslant ROI_{\text{threshold}} \text{ and Competition}(t, P(t)) \leqslant \gamma \\ 0 & \text{otherwise} \end{cases}$$

Here:

- $ROI_i(t, P(t))$ is the expected return on investment for agency i at time t given the current price
- $ROI_{\text{threshold}}$ is the minimum acceptable ROI for the agency
- $\text{Competition}(t, P(t))$ is a function estimating the level of competition based on the current time and price
- γ is a competition threshold, beyond which the agency decides not to bid due to high competition

The function $ROI_i(t, P(t))$ can be further defined based on the agency's estimation of the ad slot's effectiveness, the target audience's size, and other relevant factors. On the basis, ROI can be understood as:

$$ROI = \frac{Income}{Spending}$$

In real life the estimation of the real ROI might be more complicated than this. If an agency is conducting one advertisement campaign in one month time, the return can be of the following notation:

$$ROI = \frac{\Sigma_i(I_i \cdot N_i - E_i) - \Sigma C_{slot}}{\Sigma C_{slot}}$$

where:

- I_i is the item price for i that has been sold in this campaign duration.
- E_i is the expected sales amount of product i.
- C_{slot} is the cost of a slot.

The theoretical returns are a benchmark for the success of any agencies that is on Carry protocol. Real strategy returns can be monitored and calculated through several methods. The Carry measurement and metrics can be used for capturing the real ROIs of any efforts put on the network. Furthermore, we can adopt the P value in the statistics of the sales side to estimate the general impact of certain advertisement campaign.

There are metrics that can also be taken into consideration which agencies could measure when trying to estimate the expected outcome for certain collection of slots:

- **Total Crypto Assets of the Slot Owner's Address:** This reflects the financial strength of the slot owner. Agencies can use this information to assess whether the slot owner has enough resources and motivation to maintain and promote their slots, indirectly affecting the advertisement's effectiveness.
- **Fair Market Price of the Attached NFT to the Slot** This indicates the value of the NFT attached to the slot. High-value NFTs may attract more user attention, potentially increasing the advertisement's exposure and effectiveness. Agencies can use this information to select slots with high-value NFTs for ad placements.
- **Active User Count of the Onchain Game Associated with the Slot:** This reflects the game's activity level and user base size. A higher number of active users means more opportunities for ad exposure. Agencies can prioritize games with a large active user base for their ad placements.

In addition to the above metrics, agencies can choose from a wider range of dimensions available through the Open Carry Advertisement Network, such as user geographic locations, interests, consumption habits, and more. This allows agencies to offer more precise and personalized ad placement services to advertisers, improving the advertisement's conversion rate and ROI.

By considering a comprehensive set of relevant metrics, agencies can more accurately estimate advertisement effectiveness, providing more valuable and competitive services to advertisers.

5.3.3 Generalized first-price auction

In a GFP auction(Generalized First Price), advertisers bid based on their valuation of the ad space. The advertiser with the highest bid wins the spot and pays the bid they submitted. This auction mechanism is similar to a traditional first price auction, but GFP allows bidders to submit a variety of unconventional bids, such as maximum bids, click-through rates, conversion rates, etc.

The hallmark of a GFP auction is that it promotes competition by allowing bidders to submit their true valuation of the ad space, thus ensuring that the ad space is allocated to the most valuable advertiser. However, despite the advantages of simplicity and guaranteed revenue, GFP auctions are less stable. This is because individual advertisers may modify their placement prices frequently in order to obtain optimal revenue. For example, an advertiser may continually increase its price in order to gain display; after gaining display, it may begin to continually decrease its price in order to reduce costs. This competition is relatively arbitrary and it is easy to know competitors' bids.

In addition, when the advertiser with the highest bid stops placing, it is prone to large fluctuations in ad platform revenue. This is because other advertisers may have lower bids, resulting in a drop in ad platform revenue. Therefore, when using GFP auctions, advertising platforms need to consider how to balance the competition from advertisers and maintain a smooth revenue.

5.3.4 Generalized Second-Price auction

In a GSP auction, advertisers bid based on their valuation of the spot, and the highest bidder wins the spot and pays the price offered by the second highest ranked advertiser.

5.3.5 Vickrey–Clarke–Groves auction

VCG Auctions are sealed bid auctions of multiple items. Bidders submit bids that report the valuation of their items without knowing the bids of other bidders. The auction system allocates items in a socially optimal manner: it charges each individual for the losses they inflict on other bidders. It incentivizes bidders to bid their true expectations by ensuring that each bidder's best strategy is to bid the true valuation of the item.

The first step in the VCG mechanism is for participants to submit their preferences for resources or items. This may include valuation of different resources, order of preference, or other relevant information. We analyze data from the Carry system to obtain relevant data about the published Slot (including TVL of the game, number of viewers, number of clicks in the past, average revenue in the past, etc.) to calculate the evaluation price of the Slot as the reference price of the Slot. This reference price affects the valuation of the advertiser participants. Based on the participants' preferences and bids, the impact of each possible resource allocation scheme on the overall social welfare is calculated. The social welfare can be expressed in the form of a weighted sum where the utility of each participant is weighted by its weights.

The allocation scheme that maximizes the impact on overall social welfare is chosen as the final resource allocation outcome. Typically, this process involves evaluating all possible allocation options to determine which one is most favorable to overall social welfare. For each participant, the marginal contribution to overall social welfare is calculated, i.e., the amount by which overall social welfare would change if the participant did not participate. This is referred to as the VCG payment.The calculation of the VCG payment usually entails calculating the amount by which overall social welfare would change if the participant were to withdraw.

For any set of auctioned items $M = \{t_1, \ldots, t_m\}$ and any set of bidders $N = \{b_1, \ldots, b_n\}$, let V_N^M be the social value of the VCG auction for a given bid-combination. That is, how much each person values the items they've just won, added up across everyone. The value of the item is zero if they do not win. For a bidder b_i and item t_j, let the bidder's bid for the item be $v_i(t_j)$. The notation $A \setminus B$ means the set of elements of A which are not elements of B.

A bidder b_i whose bid for an item t_j is an overbid, namely $v_i(t_j)$, wins the item, but pays $V_{N\setminus\{b_i\}}^M - V_{N\setminus\{b_i\}}^{M\setminus\{t_j\}}$, which is the social cost of their winning that is incurred by the rest of the agents.

Indeed, the set of bidders other than b_i is $N \setminus \{b_i\}$. When item t_j is available, they could attain welfare $V_{N\setminus\{b_i\}}^M$. The winning of the item by b_i reduces the set of available items to $M \setminus \{t_j\}$, so the attainable welfare is now $V_{N\setminus\{b_i\}}^{M\setminus\{t_j\}}$. The difference between the two levels

of welfare is therefore the loss in attainable welfare suffered by the rest of the bidders, as predicted, given the winner b_i got the item t_j. This quantity depends on the offers of the rest of the agents and is unknown to agent b_i.

The winning bidder whose bid is the true value A for the item t_j, $v_i(t_j) = A$, derives maximum utility

$$A - \left(V^{M}_{N\setminus\{b_i\}} - V^{M\setminus\{t_j\}}_{N\setminus\{b_i\}}\right).$$

Additional, hesitation VCG mechanism calculation process is too complicated, we refer to Facebook's VCG mechanism. the Slot reference price of VCG mechanism will be related to CPM (Cost Per Mille) and CTR (Click-Through Rate). Because advertisers' revenue is impossible to know, eCPM is used as an alternative to CPM. eCPM is commonly used to measure the effectiveness of advertisements in an ad network, and it calculates the average revenue generated by an advertisement per 1,000 displays, regardless of whether or not the advertisement is clicked. The eCPM is calculated by dividing the ad revenue by the number of times the ad is displayed and multiplying by 1000. eCPM is given by the formula:

$$eCPM = \frac{Total\ Revenue}{Impressions} \times 1000$$

Where Total Revenue represents the total revenue of the advertisement and Impressions represents the number of times the advertisement is displayed. Then calculate

$$\begin{aligned} CPM\ Paid = {} & Next\ Highest\ Slot\ Bid * (Slot\ eCPM - \ Following\ Slot\ eCPM) \\ & + Following\ Slot\ Bid * (Following\ Slot\ eCPM - Subsequent\ Slot\ eCPM). \end{aligned} \tag{7}$$

The calculation method of deduction, if we strictly follow th e definition of VCG, we need to calculate the revenue loss of all other advertisers all over again, this is a two-tier FOR loop on the line, when there is a lot of depth of ads, this calculation is costly, we only take the 2nd ads under the first price of the first ad position to calculate this loss.

6 PROTOCOL IMPLEMENTATION

6.1 Carry Slot Contract

Carry Protocol manages game advertisements through slots. The protocol itself serves as the smart contract which manages the lifecycle of advertisement contents and slots.

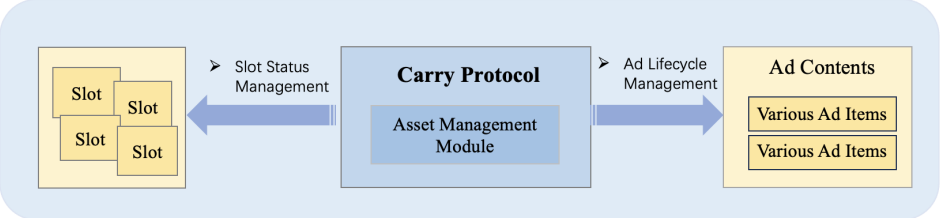

Figure 7: Slot is the basic instrument for Carry Protocol to manage advertisement.

6.1.1 Slot Creation

When a new ad campaign is about to start in a web3 game, the first step is setting up a special ad spot, known as a "slot." This slot is made into a unique digital item, like a one-of-a-kind collectible, which means it's owned and can be traded. Game makers weave these slots into their games, and then either they or the players can activate them, turning them into owned digital assets. It's similar to creating a brand-new collectible item that's ready to be paired with an ad.

6.1.2 Slot Release

A slot, once used, may need to be released or made available for future advertisements. This involves the termination or expiration of its current content, making the slot a blank canvas once more. While this could technically be a part of the slot contract, the process's simplicity might not necessitate a full-fledged contract on its own. Instead, it could be an operation under the broader slot management system.

6.1.3 Slot Content Change

Over time, advertisements or the content within a slot might need updating or altering. This could be due to changing marketing strategies, game narratives, or even player preferences. The Carry Slot Contract allows for dynamic content modifications, ensuring that the slot remains relevant and engaging. However, it is challenging to alter the embedded contents since slots are mined as valuable assets which are recorded onto blockchain, which exhibits extreme immutability. Therefore, we need to trade off between modifiability and immutability in Carry [14]. To this end, we introduce Chameleon Hash algorithm to achieve a redactable transaction property for Carry. The structure is illustrated in Figure8 [15].

(1) **Bilinear Pairing.** Let $\mathbb{GF}(p)$ denote a finite field, where p is a large prime number. We consider an elliptic curve $E_p(a, b)$ over $\mathbb{GF}(p)$, which can be defined as the set of ordered

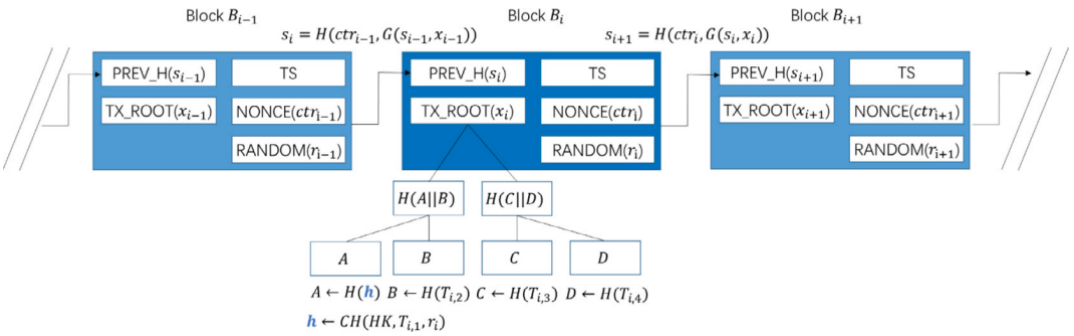

Figure 8: Chameleon Hash-based Alterable Slot Contents Solution.

pairs $(x, y) \in \mathbb{GF}(p) \times \mathbb{GF}(p)$ that fulfill the condition $y^2 \equiv x^3 + ax + b \pmod{p}$, given that $a, b \in \mathbb{GF}(p)$ and $4a^3 + 27b^2 \not\equiv 0 \pmod{p}$. This leads to the construction of an additive cyclic group, denoted $\mathbb{G}_1$, and a multiplicative cyclic group, $\mathbb{G}_2$, both of which have the prime order p. These groups are constituted of all points residing on the elliptic curve, complemented by the point at infinity. A random generator of $\mathbb{G}_1$ is designated as P. We define a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ (known as a type-1 pairing) that adheres to the following three properties [16]:

- **Bilinearity:** For any $X, Y \in \mathbb{G}_1$ and any $a, b \in \mathbb{Z}_p^*$, it holds that $e(aX, bY) = e(X, Y)^{ab}$.
- **Non-degeneracy:** If we designate $1_{\mathbb{G}_2}$ as the identity element of $\mathbb{G}_2$, it should be ensured that $e(X, Y) \neq 1_{\mathbb{G}_2}$ for any $X, Y \in \mathbb{G}_1$.
- **Computability:** For any $X, Y \in \mathbb{G}_1$, the value $e(X, Y)$ can be efficiently computed.

(2) **Complexity Assumptions.** Two fundamental problems play a crucial role in ensuring the robustness of our protocol. These problems [17, 18], known as the Discrete Logarithm Problem (DLP) and the Computational Diffie-Hellman Problem (CDHP), are outlined below: Given (g, y), where $y \in \mathbb{G}$, the advantage for any probabilistic polynomial time (PPT) adversary to find an integer $x \in \mathbb{Z}_q^*$ such that $g^x = y$ is negligible. Given (g, g^x, g^y), where $x, y \in \mathbb{Z}_q^*$, the advantage for any PPT adversary to find an element $g^{xy} \in \mathbb{G}$ is negligible.

(3) **Initialization.** The initialization phase is mainly to initialize the parameters of the elliptic curve cryptosystem and construct the public and private key information.

- The protocol generates elliptic curves E of order q, generator P, a cyclic additive group $\mathbb{G}$, and the large integer group $\mathbb{Z}_q^*$.
- Then, it generates random number SK from $\mathbb{Z}_q^*$ as the private key, then calculates $PK = SK \cdot P$, and constitutes its public and private key information (PK, SK).
- Moreover, it selects a Chameleon Hash function $H : CH(g, h, r, t)$, three one-way secure hash functions that $H_1 : \{0,1\}^* \to \mathbb{Z}_q^*$, $H_2 : \{0,1\}^* \times \mathbb{G} \to \mathbb{Z}_q^*$, and $H_3 : \mathbb{G} \times \{0,1\}^* \times \mathbb{G} \times \{0,1\}^* \to \mathbb{Z}_q^*$.
- Finally, Carry protocol publishes $\{E, \mathbb{G}, P, q, H, H_1, H_2, H_3, PK\}$ as public parameters in the ecosystem.

(4) **Construction.** The DLP is leveraged to construct the chameleon hash function. First, the entity who wants to change the contents in the slot generates a random value $SK_u \in \mathbb{Z}_q^*$ from the group of integers as his/her private trapdoor key, and the group $\mathbb{G}$ generates a trapdoor to create the public key g. The public trapdoor key h is obtained by computing $h = g^{SK_u}$.

A random value r is generated at time T_u, where $r \in \mathbb{Z}_q^*$. The value of the Chameleon Hash function T_u is obtained by calculating H. The construction and calculation of H_2 is given by the following equation:

$$H = CH(g, h, r, T_u) = \left(g^{T_u}\right) \cdot h^r = g^{T_u + (SK_u) \cdot r} \tag{8}$$

Since changing the random value r will definitely lead to a change in the chameleon hash function value, the hash value of the current time T_n is calculated by Equation (4) when T_u becomes T_n. In order to make the hash function value of time T_n and that of time T_u the same, i.e., Equation (5) holds, a new random value r' can be calculated by Equation (6). So far, the entity can calculate different time according to the corresponding random value, and thus generate the same hash value.

$$h_n = CH(g, h, r', T_n) = \left(g^{T_n}\right) \cdot h^{r'} = g^{T_n + (SK_u) \cdot r'} \tag{9}$$

$$\begin{aligned} h_n = H \Rightarrow CH(g, h, r', T_n) &= CH(g, h, r, T_u) \\ \Rightarrow g^{T_u + (SK_u) \cdot r} &= g^{T_n + (SK_u) \cdot r'} \end{aligned} \tag{10}$$

$$r' = forge(SK_u, r, T_n, T_u) = \frac{T_u - T_n}{SK_u} + r \tag{11}$$

Thus, by employing the aforementioned theoretical foundation and algorithmic principles, we can modify the contents of a specific slot.

6.1.4 Slot Advertisement Placement

The ultimate purpose of a slot is to showcase advertisements, a process termed 'placement' within the Carry Protocol. It's where the rubber meets the road. Once an advertisement is place into a slot, the slot illuminates, fulfilling its purpose. Placement could involve various parameters like the duration of the advertisement, its visual aesthetics, and interactivity levels, all governed and managed by the Carry Slot Contract.

In essence, the Carry Slot Contract is the backbone of the slot system within the Carry Protocol. It ensures that every slot's lifecycle, from creation to placement, is managed efficiently, transparently, and in harmony with the game's environment and the players' expectations.

6.2 SDK Implementation for Carry Protocol

Carry's Modular GameFi Infrastructure Platform is a significant advancement in the game industry, especially with the recent integration of asset trading into games. This platform assists stakeholders in adapting to the rapidly changing game landscape.

At its core, Carry's game infra includes an extensive data management system. This system allows game operators to collect, analyze, and utilize real-time data crucial for the game's health and growth. Operators can access detailed metrics such as user retention, online hours,

asset generation, consumption, and transactions. This level of insight is vital for making informed decisions to adjust game strategies, balance game economies, and enhance player engagement. The platform also provides tools for analyzing on-chain data, offering operators a comprehensive view of how in-game assets are distributed and used within the blockchain ecosystem.

6.2.1 Overview

We introduce a brief SDK implementation roadmap for the proposed Carry Protocol.

- Setup and Dependencies: Initialize the Carry SDK with dependencies on selected blockchain libraries and cryptographic protocols. Moreover, architect the SDK to include modules for Asset Management, Identity Management, and Security Guarantee, ensuring modularity and ease of integration.
- Module Development: Develop each module according to Section 3, implementing the formalized processes and integrating with blockchain networks and cryptographic services.
- API Design: Create a comprehensive set of APIs for game developers, covering all functionalities provided by the SDK, with clear documentation and examples.
- Integration Tools and Libraries: Provide tools and libraries for seamless integration of the SDK into existing game development workflows, supporting both Web2 and Web3 environments.
- Security Audits and Testing: Conduct extensive security audits and testing of the SDK, including unit tests, integration tests, and penetration tests to ensure robustness and security.
- Deployment and Support: Deploy the SDK for public use, providing detailed documentation, developer guides, and technical support for game developers integrating blockchain features into their games.

This comprehensive approach ensures the Carry SDK provides a robust infrastructure for integrating blockchain technology into traditional games, enhancing asset management, identity verification, and security across the gaming ecosystem. When it comes to Asset Transactions, Carry provides solutions for NFT Marketplace and auctioning. Recognizing the importance of liquidity in asset trading, it also supports integration with prominent platforms like PancakeSwap and OpenSea, offering aggregation services that facilitate trading across various marketplaces.

6.2.2 Asset Management SDK

To deploy Carry SDK, developers use JSON with HTTP to realize documentation files API as below. Specifically. we design nine common APIs for Assets Management SDK in Carry Protocol:

1. **Get FT Assets in Game**
    - Description: Searches for a user's fungible token (FT) assets within a game.

- Implementation: A GET request to /ftToken/getAssets?appId={%d}&uid=%d with game ID and user ID as parameters. This API call fetches the user's in-game FT assets, detailing each asset's name, balance, and frozen balance.
- Response Example: Returns a JSON object listing FT assets, including names and balances, to help developers track and manage in-game currency distribution among users.

2. **FT Deposit**
   - Description: Notifies the game of an FT asset deposit.
   - Implementation: A POST request to /ftToken/deposit with details including the game ID, a digital signature, and deposit information (game coin name, amount, transaction hash, and user ID). This API records the deposit transactions of FT assets into the game, crucial for updating user balances.
   - Response Example: Provides confirmation of the deposit process, including transaction status, to ensure accurate record-keeping of asset inflow.
3. **FT PreWithdraw**
   - Description: Handles the pre-withdrawal process for FT assets, freezing the corresponding assets.
   - Implementation: A POST to /ftToken/preWithdraw including game ID, signature, and withdrawal information. This step is critical for securing assets before completion of the withdrawal process, ensuring that transactions are reversible until finalized.
   - Response Example: Returns an order ID and status for the pre-withdrawal request, indicating successful freezing of assets.
4. **FT Withdraw**
   - Description: Officially processes the withdrawal of FT assets, which might involve deleting or storing corresponding assets.
   - Implementation: A POST request to /ftToken/withdraw with the game ID, signature, and specific withdrawal details. This API is used to finalize the withdrawal process, affecting the user's asset balance within the game.
   - Response Example: Confirms the withdrawal with an order ID and status, updating the asset's state as per the user's request.
5. **Get NFT Assets in Game**
   - Description: Searches for a user's non-fungible token (NFT) assets within a game.
   - Implementation: A GET request to /nftToken/getAssets?appId={%d}& uid={%d}&page={%d}&pageSize={%d}, fetching a list of NFT assets owned by a user, including details like token ID, equipment ID, and whether the asset is frozen.
   - Response Example: Returns detailed information on NFT assets, facilitating the tracking and management of unique in-game items.
6. **Get Single NFT Asset Detail**
   - Description: Retrieves details on a specific in-game NFT asset owned by a user.

- Implementation: A GET to /nftToken/getAssetDetail?appId={%d}&equipment id={%s} provides granular details on a particular NFT, including its game asset name, equipment ID, and attributes.
- Response Example: Delivers a comprehensive overview of an NFT's characteristics, aiding in asset valuation and utilization.

7. **NFT Deposit**
   - Description: Informs the game of an NFT asset deposit.
   - Implementation: A POST request to /nftToken/deposit with game ID, signature, and deposit details. This API is essential for recording the entry of NFT assets into the game, updating ownership records.
   - Response Example: Acknowledges the deposit of NFT assets, providing data such as transaction hash and order ID for verification.
8. **NFT PreWithdraw**
   - Description: Manages the pre-withdrawal stage for NFT assets, freezing the assets in preparation.
   - Implementation: A POST to /nftToken/preWithdraw submits pre-withdrawal data, securing the asset before the final withdrawal.
   - Response Example: Offers preliminary confirmation of the withdrawal request, including freezing the asset and generating an order ID.
9. **NFT Withdraw**
   - Description: Officially processes the withdrawal of NFT assets, with potential deletion or restoration.
   - Implementation: A POST to /nftToken/withdraw with withdrawal specifics, effectively changing the ownership or state of the NFT within the game environment.
   - Response Example: Confirms the successful withdrawal of NFT assets, updating records with the transaction's outcome and finalizing the asset's status.

These detailed API implementations within the Carry Protocol's Asset Management Module provide a robust framework for managing both fungible and non-fungible assets, ensuring seamless integration, tracking, and manipulation of in-game currencies and items.

REFERENCES

[1] Nikos Leonardos, Stefanos Leonardos, and Georgios Piliouras. Oceanic games: Centralization risks and incentives in blockchain mining. In *Mathematical Research for Blockchain Economy: 1st International Conference MARBLE 2019, Santorini, Greece*, pages 183–199. Springer, 2020.

[2] Herbert Hovenkamp. Antitrust and platform monopoly. *Yale LJ*, 130:1952, 2020.

[3] Valerio Stallone, Martin Wetzels, Dominik Mahr, and Michael Klaas. Enhancing digital advertising with blockchain technology. *Journal of Interactive Marketing*, page 10949968231185543, 2023.

[4] Vero Vanden Abeele, Katta Spiel, Lennart Nacke, Daniel Johnson, and Kathrin Gerling. Development and validation of the player experience inventory: A scale to measure player experiences at the level of functional and psychosocial consequences. *International Journal of Human-Computer Studies*, 135:102370, 2020.

[5] Barbara Guidi and Andrea Michienzi. Social games and blockchain: exploring the metaverse of decentraland. In *2022 IEEE 42nd International Conference on Distributed Computing Systems Workshops (ICDCSW)*, pages 199–204. IEEE, 2022.

[6] Yiming Lai, Sizheng Fan, and Wei Cai. Quantitative analysis of play-to-earn blockchain games: A case study of axie infinity. In *2023 IEEE International Conference on Metaverse Computing, Networking and Applications (MetaCom)*, pages 250–257. IEEE, 2023.

[7] Ethan D Trotz. The times they are a changin': Surveying how the howey test applies to various cryptocurrencies. *Elon L. Rev.*, 11:201, 2019.

[8] Oana Marin, Tudor Cioara, Liana Toderean, Dan Mitrea, and Ionut Anghel. Review of blockchain tokens creation and valuation. *Future Internet*, 15(12):382, 2023.

[9] Jon Truby. Fintech and the city: Sandbox 2.0 policy and regulatory reform proposals. *International Review of Law, Computers & Technology*, 34(3):277–309, 2020.

[10] Kaixin Lin, Jiajing Wu, Dan Lin, and Zibin Zheng. A survey on metaverse: Applications, crimes and governance. In *2023 IEEE International Conference on Metaverse Computing, Networking and Applications (MetaCom)*, pages 541–549. IEEE, 2023.

[11] Drummond Reed, Manu Sporny, Dave Longley, Christopher Allen, Ryan Grant, Markus Sabadello, and Jonathan Holt. Decentralized identifiers (dids) v1. 0. *Draft Community Group Report*, 2020.

[12] Qi Feng, Kang Yang, Mimi Ma, and Debiao He. Efficient multi-party eddsa signature with identifiable aborts and its applications to blockchain. *IEEE Transactions on Information Forensics and Security*, 18:1937–1950, 2023.

[13] Zhipeng Wang, Stefanos Chaliasos, Kaihua Qin, Liyi Zhou, Lifeng Gao, Pascal Berrang, Benjamin Livshits, and Arthur Gervais. On how zero-knowledge proof blockchain mixers improve, and worsen user privacy. In *Proceedings of the ACM Web Conference 2023*, pages 2022–2032, 2023.

[14] Tao Ye, Min Luo, Yi Yang, Kim-Kwang Raymond Choo, and Debiao He. A survey on redactable blockchain: Challenges and opportunities. *IEEE Transactions on Network Science and Engineering*, 2023.

[15] Chunhui Wu, Lishan Ke, and Yusong Du. Quantum resistant key-exposure free chameleon hash and applications in redactable blockchain. *Information Sciences*, 548:438–449, 2021.

[16] Palak Bagga, Ashok Kumar Das, and Joel JPC Rodrigues. Bilinear pairing-based access control and key agreement scheme for smart transportation. *Cyber Security and Applications*, 1:100001, 2023.

[17] V Jalaja, GSGN Anjaneyulu, and L Narendra Mohan. New digital signature scheme on non-commutative rings using double conjugacy. *Journal of Integrated Science and Technology*, 11(2):471–471, 2023.

[18] Mahdi Mahdavi, Sahar Khaleghifard, and Zahra Ahmadian. New variations of discrete logarithm problem. *ISeCure*, 15(3), 2023.