



Technology & Innovation



Technology Abstract

Version 0.1.9
May 2018

Abstract

We're building a blockchain-based **Global Referral Network** (GRN) which enables **Social Sourcing** - incentivised activation of result-driven organic virality.

Social sourcing means mobilising the human network towards fulfilment of a predefined result, where the network is incentivised with money and reputation to efficiently spread the word, i.e. to optimally spread the network space for a target audience which is able and interested to produce the result.

The basics of our solution enable contractors (e.g. A Yoga teacher) to generate social sourcing contracts for generating some type of results (e.g. new students, new customers, new content views etc.), while specifying the total per-result-reward offered for helping to generate each result. These contracts can then be shared with, or discovered by, some sourcing-seed audience (e.g. A current customer-base) and from there on anyone who was sent the contract and wishes to become an influencer can send the contract onwards via any means available (they're basically sharing links). Our system tracks the sharing anywhere online and enables influencers to generate complex referral maps (e.g. unidirectional referral graphs), made from atomically unique referral chains. Once a referral chain concludes in a prospect which converts (fulfils the contract, e.g. pays for a place in the upcoming Yoga course), the referral reward for that conversion is back propagated - disbursed - among the referral chain influencers for that specific converting chain.

The system itself is distributed, but allows for 2key to act as trustee, as which we may dynamically control (throughout the time and influencers space) the following parameters: **referral quota** (quota for signing-on new influencers, facilitated by a dedicated utility token called ARC - Action Referral Coin), **referral cost** (cost for referring the contract onwards), **projected reward**. The system also monitors, aggregates, publishes and acts upon inter-feedback between all players, and produces a **public reputation score** (on a hierarchic 4 level taxonomy of ~1K categories) for each player in each context (contractor/influencer/converter).

Our main technological advancements:

Game-Theory AI

Our primary focus in terms of algorithms IP is to develop a new branch of algorithms that fuse Game Theory and Machine Learning. We call it Game-Theory AI. We're already in discussions with world-leading scientists in the domains of Algorithmic Game Theory (Gideon Nave, Ron Berman, Sigal Oren, Yuval Emek, Reshef Meir) to help us charge forward in solving a general incentive and compensation model for online sharing. Moving the web to a general proactivity incentive model is a big step, and requires developing a dedicated technological framework to enable dynamic, real-time, personalized, action-type-tailored incentive-compensation models to ensure all players maximise their reputation and returns. Heretofore, Game Theory and Machine Learning haven't actually been fused, mainly due to lack of a real world framework and the resulting big-data sets allowing to develop general incentive models for online value-generating proactivity. The 2key RefNet will produce, for the first time, both the framework and the datasets to enable conducting research and development of Game Theory AI models for online sharing. We're looking to lead the front in integrating Machine Learning and Game Theory to implement generic action-referral-reputation dynamic-incentive models to optimise returns for all players and efficiency of the system at large. The system needs to optimise for minimal time/steps to conversion, maximal number of conversions, maximum amassed reputation by all participating players, minimal abuse, and optimal price point (reputation-money mix) between contractor and influencers (in that final regard we could also help the contractor frame the initial reward price point).

Token-Economics AI

Strategically, the economy will be managed via a GRN foundation smart contract that utilises novel **Economics-AI** to ensure optimisation of the RefNet's economic viability KPIs, namely weight (market cap X users) and flow (daily transactions X avg transaction size). The Economics AI will control tuning parameters manipulating the rates and usage channels of the ICO's marketing pool of tokens (e.g. specification and release of economy-wide reputation rewards, of 2key self-marketing contracts etc..). Periodic selective reputation-based rewards will act to promote the users actually contributing to the economic function of the network, thereby incentivising the agents empowering the network's viability. By gradually allowing more levels of control on a network-wide scope, such AI can optimise for global network KPIs to keep the interests of the economy.

Referral-Recommendation-Graph

We're looking to develop a referral-recommendation graph to facilitate match-making between recommendation seekers and valid recommendation givers. This can empower the refnet by feeding influencers with potential high quality referral targets from within their networks, thereby lowering friction and removing barriers, empowering the building of more efficient referral maps. Solving this requires developing a novel fusion of methodologies from the domains of Natural Language Processing, Natural Language Understanding, Unsupervised Machine Learning, Graph Theory and Social Sciences.

Pixeless Tracking

Our primary backend novelty is to enable purely online pixeless-tracking, i.e., we're affectively migrating the internet's tracking from siloed pixel integrations within business websites or apps into the links themselves. Currently, there's an internet-wide reliance on pixels for marketing-attributions and conversion tracking. Pixels require complex integrations and ongoing management by site/app owners, they have no visibility outside the websites/apps in which they're installed and there's effectively no inter-net tracking (i.e. the tracked info is segregated and owned by separate competing website/app owners). With 2key's solution, there's no technical overhead on business sites and zero integration requirements - since the tracking itself is performed by the links being shared, and not by the endpoint site/app the links are pointing to, as are the conversion events, which can be produced via the smart contracts activating the links. This will enable 2key to offer a fully SaaS CPA model with multi-step-multi-chain referral reimbursement support triggered by actual conversions/acquisitions (enabled via the smart-contract tokens/links themselves). This has the potential to fully solve the holy grail of marketing - a full attribution model, where multi-step referrals are inherently charted by the mere passing of web3.0 tokens between consumers, and these same coins are also used for producing the conversion events. This produces a closed loop blockchain based technology which enables businesses to open conversion contracts, and have these shared seamlessly online, tracking the full referral chains, and reimbursing those who helped effectively spread the word (i.e. market the business's offering) upon successful conversions. Since we're committed to always being in the market with a product ready for mass-adoption, we're also rolling out full web2.0-3.0 integration support, allowing users with no web3.0 client, to also use our services with minimal UX overhead via our dedicated web2.0 interfaces for the 2key smart contracts, and our managed wallets solution which doesn't require any app/extension/plugin install, and is interactable directly from any existing web2.0 browser.

Web3.0 Infrastructure

In order to produce the GRN, we're developing novel, patent pending solutions in terms of web3.0 infrastructure, mainly:

ARCs - Action Referral Coin

On the blockchain infrastructure front, our novel patent pending tracking technology is integrated into the ARC, a new kind of blockchain infrastructure allowing for links that play out smart-contracts and perform self-tracking and management as they are distributed online. ARCs allow complex, multi-step, multi-party, state syncing and conversion events, as well as continuous real-time access to a blockchain-synced state, enabling continuous dynamic monitoring, moderation and analytics of the smart contract network of operation, available per user-role permissions in the contract.

ARCs will hold a central role in our architecture, as their Game-Theory AI-determined balance shapes the topology of the referral maps for each contract. Beyond their algorithmic role, they need to be engineered in a novel fashion, to allow them to act as web3.0 coins on one hand, while facilitating their true function of charting referral chains, showcasing the offered services or the required contract result, and enabling the fulfillment of the contract via any conversion event (leaving details, giving info, paying for products or services, viewing content etc.). Each 2key contract holds a balance of ARCs, which are the tokens actually being distributed by influencers. The infrastructure for their interplay with each other and with the 2key contract they are linked to is at the heart of our novel architectural solution.

Multi-Party State Channels

In order to facilitate a truly decentralised offering, while enabling scale, configurable transaction costs and permission-based access to contracts, we're developing multi-party state channels. We plan to release the basics of the solution as open source to allow any DApp requiring asymmetric user roles in smart contracts to utilise this development. In such cases where user roles and permissions to determine consensus in the contract are asymmetric, we can mitigate the harder consensus problem in general purpose multi-party state channels, (a problem which explains their current absence from web3.0 infrastructure), while still providing a working solution which enables actual multi-party state channels with no star-formed architecture holding a central server as the middleman. The main novelty behind our solution here will be to use pseudo-symmetric roles between state channel players, giving an originator (e.g. a contractor) ,the ability to name a single player as consensus moderator (e.g

2key), thereby empowering a single-hot connection between an end-user and a web2 interface for a contract, or the ARC token being sent to an influencer's wallet in the pure web3 approach, as minimally sufficing events to link these influencers into the multi-party state channel hosting the contract. Whether it be a cookie or other browser code playing out in web2, or the ARC contract code playing out as it passes through an account's wallet in web3, being still linked to its parent 2key contract facilitating the channel, we're aiming for a solution which is seamless for any user.

Multi-Party State Networks

For scalability purposes, we aim to develop the ability to deploy our full solution on an Ethereum mirror blockchain, facilitated by an ethereum node cluster which is served by 2key Ltd., so that we can offer full price per share control by the incentive models, and not be limited to slowness or gas prices in the main network. The syncing between the mirror net will allow however, to transfer tokens and reputations earned to become real tokens and reputation on the master node. This may be seen as an extension to the multi-part state channel approach offering a a state network which in effect stamps itself onto the main net for updating state changes, and mirrors the current state between the main and mirror blockchains, so that the more advanced state always prevails, whether it originates from the main or mirror chains. Playing out a contract can thus occur on the mirror chain, with no scalability, privacy of gas constraints, and the products (reputation + token) are synced ("pushed") to the main net allowing the GRN users to enjoy the financial and reputation security aspects of the main net, together with the privacy and scalability aspects of the mirror net.

Decentralized Web2.0 Link-Based Smart-Contracts

We have developed a patent pending fully decentralised, web2.0 pure multi-party state-network solution for scalable yet fully decentralised multi-step tracking using nothing but web2.0 links and users' browsers, while still being synced to the Ethereum blockchain for governing the underlying smart contract and ensuring security, fairness, fraud-prevention and contract-adherence.

This blockchain infrastructure layer allows actually decentralised integration of web3.0 (the Ethereum Blockchain) into web2.0. First, it allows smart contracts to play out off-chain, in a completely decentralised manner, while keeping the contract's state and users synced and reliable utilising the ethereum network, and secure to hacking attacks e.g. sybil attacks or fraud in referral chain malicious modifications utilising state of the art cryptographic methodologies.

The Contractor only pays gas for generating a contract, and a Converter only pays gas for converting a contract, which themselves can also be mitigated utilising a 2key gas station. All the referral graph is charted utilising links which are used to pull and run source code from static serving in a public repository managed by 2key Ltd., the code allows all interface with the 2key contract, and when joining a 2key contract, influencers which don't have a web3 account are associated an account managed by 2key, but the private keys themselves are stored in the private storage of their browser, similar to MetaMask.

Our solution ensures that the contract plays out via influencers who were valid and authorised to share the contract, and allows influencers to join downchain only after proving their referring chain was legitimately obtained. Once a conversion occurs, the funds for rewards and conversion funds themselves can be kept in escrow by 2key, and only then the contract deciphers the validates the refchain legitimacy, to ensure only valid influencers are rewarded for each conversion.

This model doesn't allow for dynamic incentive models to modify ARCs balance and price to share dynamically per influencer and throughout the contract lifecycle, but they do allow for dynamic attribution models to be played out upon conversion and lookback on the refchain. In other words, these types of contracts do not hold ARCs and do not utilise ARCs, so there's no real-time forward looking tracking of refchains - meaning influencers and the contractor will not be able to look forward on the refmap stemming from them, until a conversion actually takes place. Only then will the contract itself on ethereum be contacted to activate the conversion validation and reward distribution. In ARCs based contracts, the ARCs travel from influencer to influencer, and allow each influencer direct contact with the contract on each touch point. In this mode, only the thin code originally downloaded to the browser from the link's primary domain is played out, and allows influencers to generate downchain referral links to create deep layered referral chains, but the contract itself in ethereum is only interfaced at the start and the end points (conversion events) of the contract.

Web3.0-2.0 Integration

To enable a truly seamless web3 integration for web2 users, we are developing the following integrations:

Web2 smart-contract interfaces

Our web2 apps (mobile+desktop) will allow contractors to generate 2key contracts (after explicit sign in), while the 2key contract itself, once generated, will receive a dedicated web2 interface url, whereby any user can utilize any web2 browser to seamlessly interface with the smart contract (no explicit sign-in required) - to join a referral chain or to fulfill the contract.

Web2-client-agnostic hosted web3-wallets

Our web2 interface for the contracts will also support generating hosted wallets for any users signing into a contract as influencers or converters. Surpassing metamask, no extension needs be added, no plugin must be installed, but rather the web2 interface itself will implicitly generate a wallet for any user trying to interface 2key contracts without a valid web3 account. We will give each user the option to either have their private keys stored on their machine or let the 2key mainframe conduct it. Once influencers will opt to query their status via the 2key service apps, an explicit sign in will be prompted to make sure the funds stay secure.

Web2.0 Serverless Infrastructure

All our web2 infrastructures, while requiring centralized control by 2key, employ pure serverless/docker-swarm architectures, while moving as much of the code as possible to the front-end, to march towards decentralized hosting of 2key apps.

Thanks for reading!

For any questions, comments or remarks, please contact eitan@2key.co