Winding Tree Marketplace 2.0

Draft for community review. Subject to change.

Maksim izmaylov

2021-08-10

Winding Tree Decentralized Travel Marketplace 2.0

Draft for community review. Subject to change.

The idea of a decentralized travel marketplace was conceived by our team back in 2016, inspired by our experience working on other projects in travel. Later that year we published a manifesto[1], where we described the need for an intermediary-free environment, a free market controlled by its participants. We envisioned that such a platform would not just reduce fees and remove entry barriers, but also promote trade and innovation. The manifesto was well-received by the industry professionals, and we further developed our ideas in the original Winding Tree white paper[2].

We released the pilot version of the Winding Tree Marketplace in 2020, but we did not achieve the desired adoption rate. The goal of this paper is to analyze why that happened, identify concrete issues, and outline a roadmap that would make the Marketplace the ultimate travel distribution platform.

Decentralized Marketplace

Winding Tree is a decentralized B2B marketplace. In order to avoid any confusion and remove ambiguity, let's first define what it is.

The Problem

All human activity that's happening on a grand scale is facilitated by market-makers, intermediaries. A few examples come to mind:, Apple and Google App Stores, Amazon, Uber.

All these platforms solve the problem of coordination, they connect buyers and sellers. For example, an app store is where app developers and users of those apps meet. Amazon connects retailers and consumers, while Uber matches drivers and riders. Centralized design makes sense: it allows parties on both sides of the marketplace to come to one place in order to find what they need.

At the same time, centralized design is not perfect. Centralized market maker is, by definition, a monopoly, and today's monopolies are not shy to use their position to their benefit: they charge high fees, they dictate who joins the marketplace and who doesn't, etc[3]. In other words, they have total control over their turf.

Travel industry has its own market-makers: Priceline and Expedia in hospitality; Amadeus, Sabre, and Travelport in air travel; AirBnB and VRBO in vacation rentals. These companies are providing

[1] https://www.linkedin.com/pulse/travel-industrys-invisible-battle-maksim-izmaylov
[2] https://github.com/windingtree/white-paper/blob/main/white-paper.pdf
[3] E.g. Jason Fried on Apple's Policies https://hey.com/apple/iap/

tremendous value to the world, but they also have high transaction costs (15% average in hospitality, sometimes even more in other fields), they make sure entry barriers are high, and they lack incentives to innovate (that is why, according to some industry experts, travel is about 10 years behind in terms of adoption of new technologies).

Our goal was to create a decentralized market-maker that would have all the benefits of its centralized counterparts, while not allowing for any single company to control the marketplace.

The Challenge

After experimenting with different approaches, we came to the conclusion that we should focus on two main features that our marketplace should have: discovery and trust. Thus, we needed to create a decentralized identity system for companies.

Discovery

The most important service that a market-maker, decentralized or not, should provide is a way for market participants to find each other. In other words, buyers should have an up-to-date database of sellers, and vice-versa. This database can't be controlled by a single party, anyone should be able to enter their organization into it, without asking for permission. The solution to this problem is obvious: use a smart contract. It is, of course, a trade-off. Let's see why using a smart contract is a good idea.

	Traditional Database	Smart Contract
Control	Controlled by the owner of the server where the database is hosted. It means that your company's account in the database can be revoked or altered by the ultimate owner of the database.	Controlled by accounts explicitly defined in the contract, e.g. by the community of marketplace participants, which means that you control your account.
Access	Access to the database for the purpose of creating, editing, or removing records, has to be explicitly granted by the controller	Can be accessed by anyone

	Traditional Database	Smart Contract
Trust	High level of trust due to elaborate KYC processes (often a regulatory requirement)	Low level of trust, because the system is open
KYC	Slow, extensive and expensive	Not applicable: there is no one to perform KYC[4]
Availability	Determined by hosting SLA	Always available
Scalability	Trivial to scale to thousands of transactions per second	Scalability improves extremely slowly and depends on development of the whole ecosystem
Maintenance Cost	High, born by the intermediary	Low, born by the miners of the blockchain, who offset it with transaction fees
Onboarding Cost	High	Low

Trust

Smart contract is a virtual place where buyers and sellers can meet, but there is no trust in the system. There is no way to verify the validity of the information provided by its participants, and that proved to be the real challenge.

Blockchains are sometimes called “trustless” environments. Indeed, there is no single authority to instill trust, but there are strict mathematical rules that create it, if we’re talking about simple blockchain transactions like cryptocurrency transfers[5]. The problem we had to solve—mapping real-world legal entities to their blockchain-based digital identities—is a lot more complicated than that, but we knew it could be solved by applying the same principles that blockchains are based on.

Transactions

After buyer and seller have found each other, they need a way to transact. Seller needs to provide an API, and buyer should have a way to access it. At least the first few API requests must bear ORGiD-powered cryptographic signatures in order for the two parties to establish each other’s identity.

[4] KYC stands for "know your customer", KYB - know your business, AML - anti-money laundering

[5] All blockchains are different. Not all blockchains are decentralized, even if they claim that.

Settlement

Each booking has to be paid for, and thus a settlement solution is required, which would have to be integrated into a traditional banking system, for travel companies will not start using crypto currencies for a long time, although they should have that option.

ORGiD: Self-Sovereign Identity for Organizations

After extensive research and experimentation we created a specification and design for the solution that would solve the problems we outlined above. We realized that what we created is, fundamentally, an identity solution for organizations, and we called it ORGiD. ORGiD is a Self-Sovereign Identity (SSI) protocol for organizations[6], and its implementation on Ethereum[7]. ORGiD is a global, open, decentralized business registry with a built-in decentralized KYC process.

In the past, coordinating large numbers of buyers and sellers could not be done any other way but through an intermediary, simply because the cost of paper-based KYC P is extremely high. In a marketplace with m buyers and n sellers, and cost of verification $P \gg 0$, the total KYC cost for verification by an intermediary is

$$P * (m + n) \tag{1}$$

and

$$P * m * n \tag{2}$$

in case every buyer and seller verify each other independently. It's easy to see that if P is large, centralized verification is a much cheaper way to organize. Although if $P \to 0$, the total cost would also be approaching zero.

The good news is that with modern cryptography and public blockchains, the cost of establishing an identity, as well as the cost of verifying it, is indeed approaching nil, and that is why we're experiencing the SSI boom.

Issuing and verifying paper-based documents is exceedingly expensive, and the main disadvantage is that it was not designed for the internet era. Paper documents are meant to be verified by a human being and in person, but thirty years after the creation of the first web browser, it is still commonplace to send copies of our identity documents via email, which puts the sender at a great risk (who can

[6] https://github.com/windingtree/org.id
[7] https://etherscan.io/address/0x6434DEC2f4548C2aA9D88E8Ff821f387be3D7F0D

guarantee that their ID will not be leaked or misused?), and doesn't give the receiver any information at all, since electronic copies of paper documents can be easily manipulated (so there is almost no point to match photo ID with the face of the sender).

The next logical step in the evolution of identity systems could have been centralized cryptography-powered identity systems, and many jurisdictions and organizations adopted this approach. Although it seems that this phase will not be long-lived, considering the current pace of development in the field of identity, with the EU (and Germany in particular) leading the way, which signifies the failure of centralized identity systems. Cryptography-powered centralized ID systems have one deadly flow in that they violate the most fundamental presumption of public-key cryptography that states that key pairs must be generated by users themselves. Delegating this crucial task to someone else, especially to a single entity, as it is the case in Estonia[8], for example, makes the whole system vulnerable to manipulations and attacks[9].

Let's summarize. We have intermediaries because paper-based KYC/KYB/AML checks verification methods are extremely expensive. Especially so in travel, where a travel agency has to work with hundreds of different suppliers. Because of the same reasons, the intermediaries have to be gigantic, connecting thousands of businesses, which eventually leads to monopolistic behavior.

The alternative is a system where verification is performed once, or rather, where it is an ongoing process that happens every time one company interacts with another, where trust is being constantly uploaded into the system by its users, but instead of working with an intermediary, we are proposing a new way of doing business online.

Streamlined KYB

Optimizing KYB processes, drastically reducing their cost $P \to 0$, is key to unlocking tremendous value that today is wasted on ineffective processes. ORGiD aims to remove that unnecessary friction from online commerce by matching real-world legal entities to their blockchain-powered digital identities. Companies will benefit from using ORGiD because:

- The cost of issuing and verifying digital credentials linked to an ORGiD profile than to issue a paper document approaches zero;
- It virtually eliminates fraud because it's infinitely harder to steal a cryptographic keys than to forge a paper document;
- ORGiD has a free open source data standard for describing organizations: ORG.json;
- Organizations can easily provide an audit trail of a document, transaction, or any other piece of data;

[8] https://www.usenix.org/system/files/sec20-parsovs.pdf
[9] https://nakamotoinstitute.org/trusted-third-parties/

- ORGiD promotes unbounded innovation; it is open source and free to use, which allows third parties to develop new applications and services.

Free Market

What sets ORGiD apart from most other "self-sovereign" identity solutions is our uncompromising dedication to the ideals of decentralization and free market. It is frustrating to see companies that fly the colours of decentralization just to use it in their marketing strategy, while trying to insert themselves as intermediaries. In other words, it's business as usual.

Conversely, ORGiD is built to be open and free. We described in detail the reasoning behind our philosophy in our first white paper. In short, only open technologies, like computer, internet, blockchain, will see mass adoption. Luckily, there are not many people that understand that, and this is one of our advantages.

ORGiD is open source and will always remain open and free technology. It is guaranteed by smart contracts and open source licenses in its repositories[10]. ORGiD is built upon standards developed by W3C: DID (decentralized identifiers[11]) and VC (verifiable credentials[12]).

How it Works

An ORGiD profile consists of one or more records in the ORGiD smart contract[13]. Each profile may have information about the legal entity, as well as business units (or just units) of the organization.

For example, a representative of Acme Corp. may create an ORGiD profile where they would indicate Acme's legal address, its government-issued identification number, etc., HQ address, etc. They may also add Acme's many businesses: Acme Anvils (a physical store located at a certain address), Acme Online (an online store), etc. Acme also has multiple departments: back-office, accounting, IT, etc. Each of these businesses and departments may have its own designated director (Ethereum account) who is able to control unit's data. This design allows companies of different sizes to use the system.

[10] https://github.com/windingtree/
[11] https://www.w3.org/TR/did-core/
[12] https://www.w3.org/TR/vc-data-model/
[13] https://github.com/windingtree/org.id

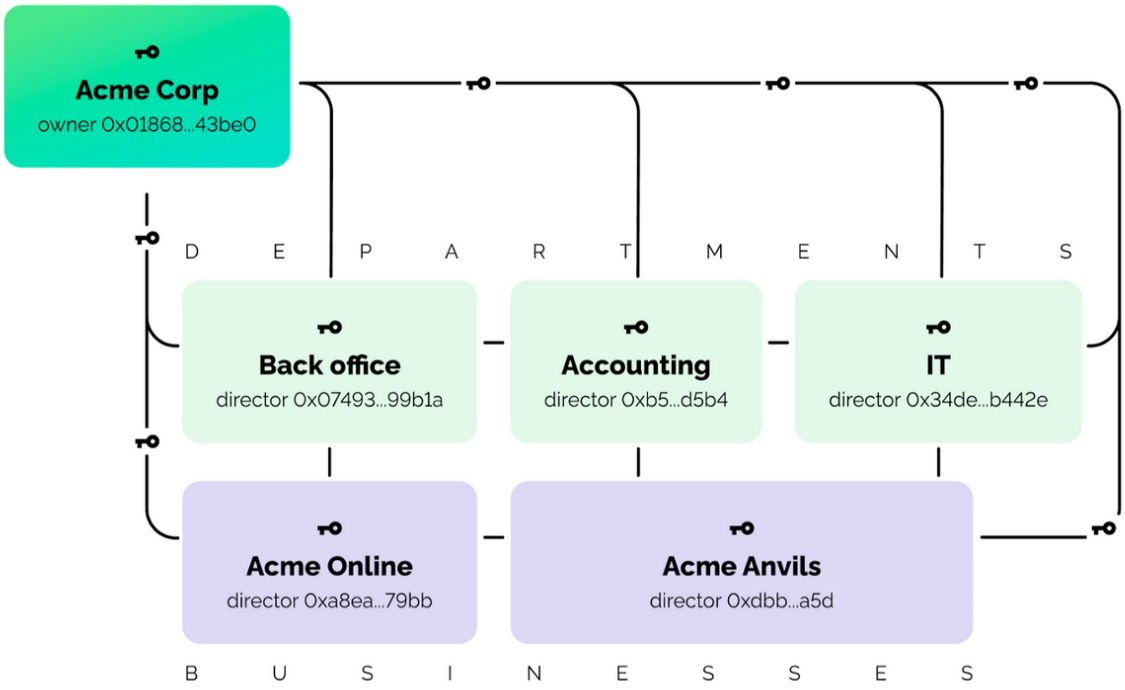

Figure 1: ORGiD profile may consist of several ORGiD records

Each of these records contains a link to a datasheet stored on IPFS[14] (optionally, on any server on the internet), which contains a description of the legal entity or one of its units in ORG.json format.

These datasheets may contain a wide range of data points about the entity it describes, including public keys of employees or representatives of the organization, as well as other software and hardware keys. These keys, in conjunction with API endpoints, also published in the datasheet allow two companies to start transacting right away. Let's look at an example.

API Access

Acme Corp owns and operates a few hotels. They have a division of the company called Acme Hotels. The datasheet for Acme Hotels has information about an API endpoint that can be used by other companies from the ORGiD ecosystem to view and book inventory from Acme Hotels, for example `/orgid-auth`.

Best Travel is a travel agency. They would like to integrate Acme Hotels inventory into their offering. Best Travel already has an ORGiD profile, where they have listed at least one public key that they intend to use in communication with other companies in the ecosystem.

[14] https://ipfs.io/

Below is an example data flow between the two companies. If supplier's software determines that buyer's ORGiD profile possesses adequate credentials for accessing the API, then it may issue an access token that would work with other API endpoints.

Note that supplier's software needs to be able to decipher incoming ORGiD-signed requests, as well as to be able to filter those requests based on the trust data contained in the datasheet. Buyer's software, in turn, must be able to speak the supplier's API language.

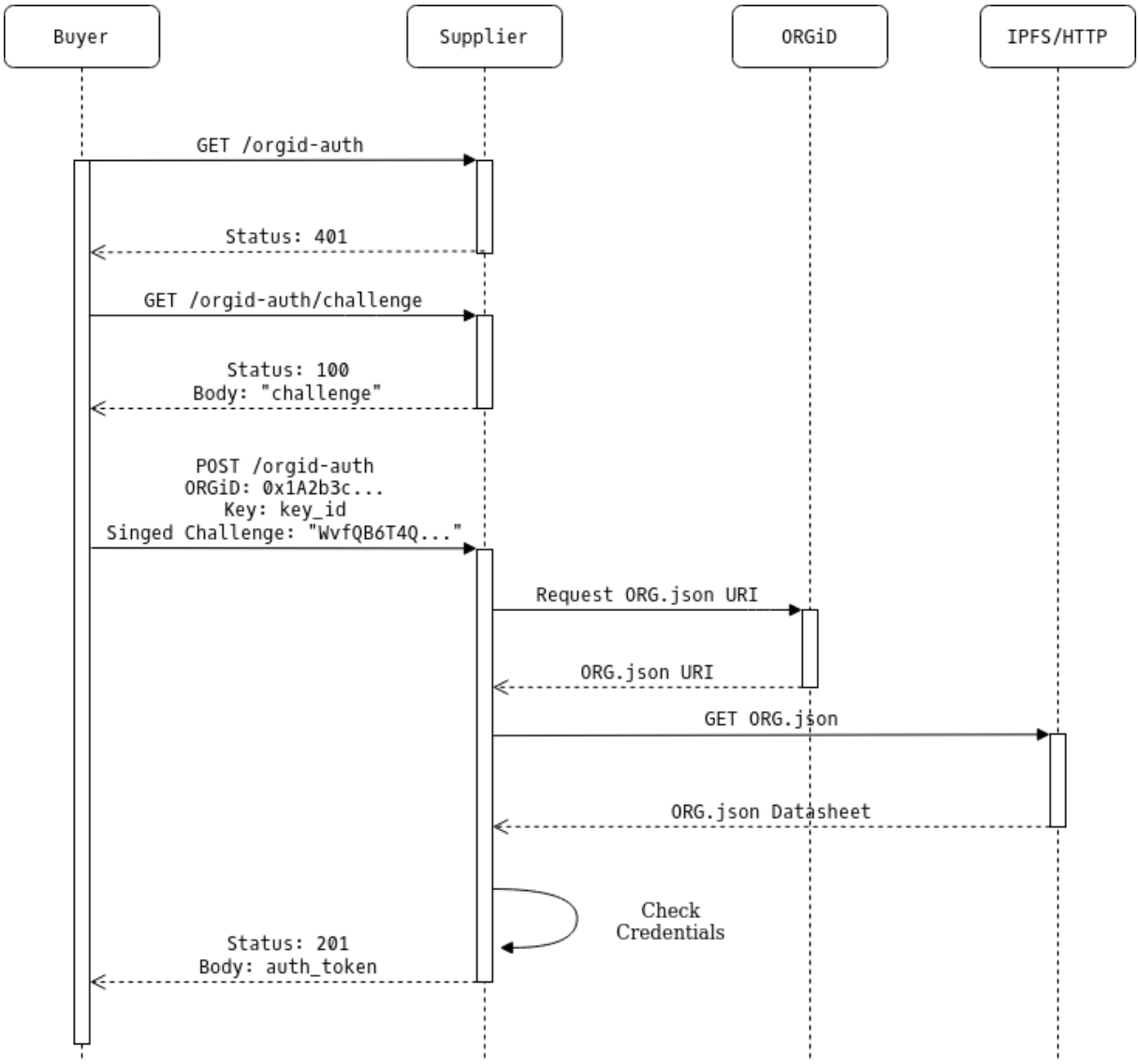

Figure 2: ORGiD authentication sequence example

Trust

ORGiD provides a mechanism for companies to be able to prove to other participants of the platform that they are who they say they are, which can be done in several stages.

Proof by Association Company links their website, social media accounts, and other virtual properties to their ORGiD profile.

Credentials and Certifications Company lists credentials provided by other companies and organizations (e.g. government-issued certificates, associations, memberships, etc.) ORGiD does not aim to replace existing KYC processes, but rather take them online. So, for example, IATA will be able to issue an IATA number to a travel agency in the form of a signed verifiable credential.

Community Vetting After completing the base profile, a company may decide to be verified by the community, which provides the highest level of trust on the platform, and also allows the company to participate in its governance.

The process for community verification was inspired by Kleros. Here's how it works.

1. The Winding Tree DAO creates appropriate Directories for different types of companies that will be trading on the platform. Directories for hotels, airlines, and travel agencies were already created by Winding Tree. Each Directory is formed around a strict set of rules that companies must follow.
2. Members of the community may choose to participate in the community vetting process. They submit a LIF stake and become the watchmen of the network with the ability to vote on adding or removing companies from Directories.
3. In order to be added to a directory, an organization representative submits a LIF stake[15]. The submission enters a pending stage, during which watchmen may submit evidence against the inclusion of the company into the chosen directory. Challenger also stakes LIF so the Dispute could be created. After the submission is challenged, a jury that consists of a few randomly selected watchmen is assembled. They review the evidence for and against the submission and issue their verdict. If the submission is not challenged within a certain period, the company is entered into the directory.

Companies that have been entered into one or more Directories will be able to participate in the governance process of the Winding Tree DAO.

[15] LIF stake may seem arbitrary here. Why not stake ETH or any other token? We should not forget that this project was funded by the community of LIF holders and this is a way for us all to give back to the people who helped make this project happen.

Winding Tree Marketplace

Arbor is the canonical open source[16] user interface to the ORGiD infrastructure. It is designed to be the baseline and inspiration for creation of other user interfaces to the ORGiD ecosystem.

Winding Tree Marketplace, the first implementation of Arbor, was launched in June 2020.

Primary Use Case

Winding Tree Marketplace was created specifically for travel companies (while ORGiD may be used by companies in any other industry). The first version of the Marketplace allowed travel suppliers to register an ORGiD account, link it to their website, instagram, etc. (thus proving to other marketplace participants that it is a real company), and then expose their APIs.

Adoption Problems

We promoted the marketplace extensively, but our hopes for the quick growth of its user base were shattered by the harsh reality. What went wrong?

UX is Too Complicated for Most

Most people in the travel industry (and in the world) are not familiar with concepts behind blockchain, with its terminology and workflows. For a person familiar with crypto it isn't too complicated:

- Install Metamask
- Create a Metamask account
- Acquire some ETH
- Log in using Metamask
- Fill out a registration form
- Provide a compatible API
- Send a transaction to the smart contract

For someone who already has a crypto wallet, all these steps are a breeze to go through. But for someone who is just starting to experiment with blockchains, these steps will take days to complete, and that is the minimum of required steps.

[16] https://github.com/windingtree/arbor-frontend and https://github.com/windingtree/arbor-backend

Just creating an Ethereum account means the user has to learn about seed phrases, how to store them securely, how to make sure the seed phrase is not compromised, and why you have to take all these precautions. It's a lot to take in at once.

Next step, getting ETH, is even more complicated, because it's a third-party dependency: the user will have to register at an exchange, probably having to go through a (paper-based) KYC process, which could take days.

Multiple Stakeholders and New Business Process

Another problem we had faced is that the decision to fully adopt the Winding Tree approach of doing travel distribution can't be made by a single person, even if it's the CEO. All the layers of management will have to understand the change they're buying into:

The purpose of listing on the Marketplace is to allow other participants of the Marketplace to book inventory automatically; the supplier will have to trust the Marketplace's KYB process.

Supplier's software will have to recognize API requests outfitted with a special ORGiD signature, which would contain the information about the company making the request.

Buyer's software should be able to send API requests with the aforementioned ORGiD signature.

Supplier's software will have to have logic for letting API requests from companies with an acceptable level of trust to go through, while rejecting the others.

Third Party Dependencies

Even if all the stakeholders understand the new process, very often it is the case that the suppliers' API is provided by a third party, who may charge for a new API integration.

Ethereum Scalability

While Ethereum is by far the most respected public blockchain, with a vibrant community and extensive tooling around it, it is still suffering from inadequate scalability. For example, creating an ORGiD account on some days, when ETH price was high, and miners were requesting high gas fees, could cost around $400 US dollars.

Marketplace 2.0

We know that the vision behind the marketplace is correct. It is constantly reinforced by travel industry's executives and experts that we talk to. How do we take the Marketplace to mass adoption?

Strategy for Seeding the Marketplace

First, we need to attract a few big players to the Marketplace. It takes a lot more time to close a deal with a big company, but the result is a lot of supply/demand on the marketplace. Depending on the size of the deal, we could spend a lot of time creating customized solutions for these partners, thus bypassing some of the problems we spoke about earlier (gas price, UX, etc.) Big companies also have resources for creating their own customized solutions and integrations.

After the Marketplace is seeded with a few large buyers and suppliers, we'll be able to open the floodgates and let the rest in, which would bring its own challenges, most of which will be solved by education and better user experience.

Better User Experience

Section in the works…

Multiple Blockchains

Blockchain transactions will never be free, that's the price of decentralization, the responsibility we have to accept if we want to be truly free. But how does a mere mortal obtain ETH and LIF to participate in the marketplace? Good news is that we have already optimized our smart contracts and reduced the gas fees by 80%. Still, if Ethereum is congested, gas price could be an insurmountable barrier for most, and the only way to solve that is to deploy our smart contracts on an L2 solution, or perhaps several of them, perhaps even allowing the user to choose which smart contract to use.

Third Party Dependencies

While we create customized solutions for big companies, we have to provide a way for small players to join the marketplace, that is the key to mass adoption. So far we've identified a segment of the travel industry that is underserved by current software providers, who will be able to change their current business processes relatively easily. For a well-established hotel brand, for example, that uses a system like Oracle, it could be expensive to onboard the new Winding Tree powered business logic.

On the other hand, a hotel with an Excel spreadsheet for managing bookings, adopting a new piece of software would probably help improve their operations. This is why, for example, we created Rooms, the open source hotel inventory management solution, which is connected directly to the Winding Tree Marketplace via its API. But now we're facing the next challenge: we have to promote and sell Rooms to hotels and compete with already existing solutions.

New Token

Section in the works…

Other Use Cases

Section in the works…

Winding Tree DAO

All community-vetted organizations are granted the right to participate in the governance of the ORGiD ecosystem via the Winding Tree DAO.

Smart contracts, like any other software, have to be updated, either to fix bugs or introduce new features. But if smart contracts are not controlled by a single person or entity, who can upgrade them? The community itself, of course, with the help of a special smart contract that is able to rewrite all smart contracts in the ecosystem (including itself), if that action is approved by the community.

Proposal Creation

Like in any other democracy, a change begins with a proposal from a community member. In the proposal they must explain the need for the change, how they think it's going to affect the whole community, the cost of implementing the change, elaborate on all the pros and cons of it, etc.

The proposing member may also write code implementing their proposed change, or ask another member of the community to do so.

After the proposal is submitted, it enters a discussion stage, during which the community will determine how much this change will affect the ecosystem, introduce new considerations into the proposal, etc.

Voting

After the community has discussed the proposal, it enters the next stage, where the members vote for or against the proposal. If the proposal is approved, the code of the smart contract in question is updated.