



Internet of People

Fermat, the Internet of People and the P2P Economy

Version 0.3 - Nov 2016

Authors

Luis Fernando Molina

Contributors

Amadeo Charlé, István Zólyomi, Daniel Csendes, Wigy

This document describes the Fermat vision, Internet of People and the P2P Economy.

The Fermat Project

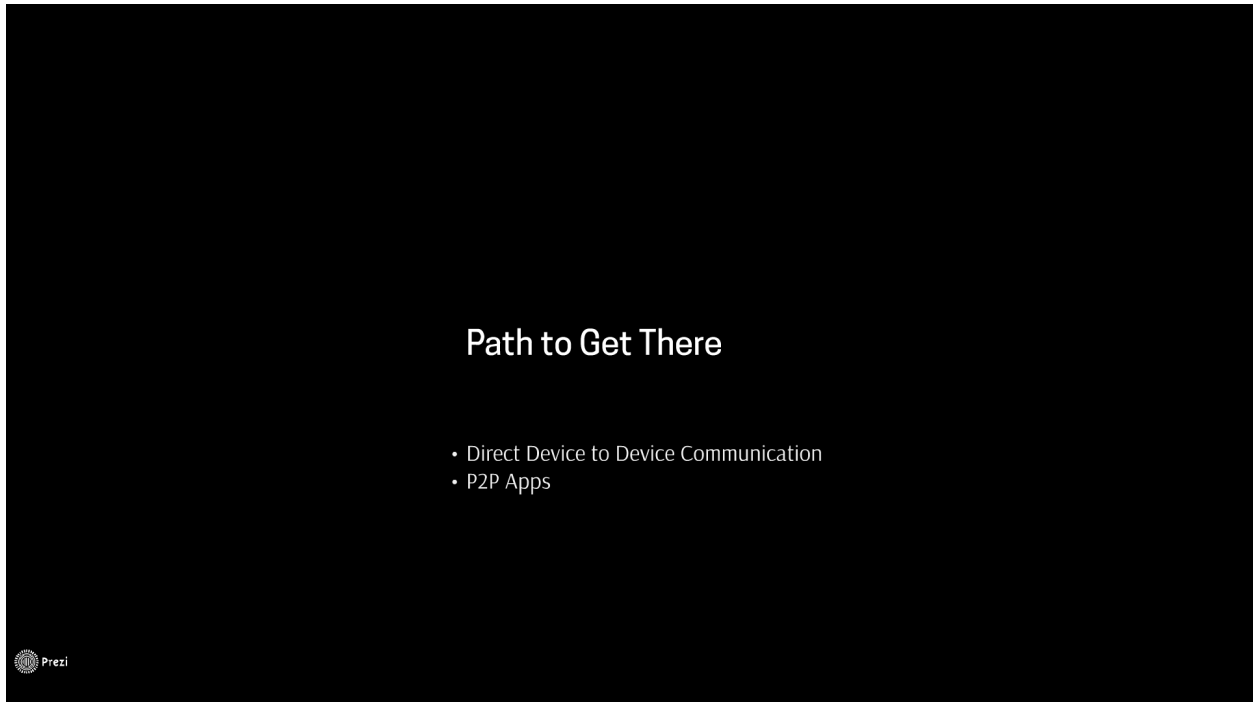


We envision a world where people can freely interact electronically without unnecessary third party interferences. Both for social and commercial interactions. No spying, no censorship, no taking a cut on private transactions between individuals, no mining of private information, no unnecessary middlemen. A world where people are more important than entities like companies and states, a world where people have the choices and the means to interact directly between each other.

We have been living inside an ongoing trend towards centralization of power and control. In present time a handful of entities (giant companies or conglomerates and some states) dictates what a huge percentage of the world population can or cannot do. In this current path is not difficult to predict that someday in the future some entity will control it all and there is when things can go really bad. Fermat is a community of people standing up against this trend, aiming to create a future where humans will be connected between them but in a decentralized way.

What we need to enable now are direct device to device communication, with data being stored at end user devices and apps built to interact with each other directly, over the Internet but without going through the web or requiring any service from any company or institution. We need P2P Apps that can run independent of any entity. This is the Fermat Project vision.

Which is the path to get there?



Today, eCommerce and electronic social interactions are dominated by a handful of tech giants. Using different technologies they have built huge private networks of individuals trapped on those networks and subject to the arbitrary rules they impose and the fees they charge upon users. People in these private networks usually have their private information mined, and their privacy exposed and potentially hacked. The path out of the status quo is the following:

1. **Direct Device to Device Communication:** We need to connect our personal devices between each other with direct connection over the internet.
2. **P2P Apps (Person to Person Apps):** We need Apps for our personal devices that know how to speak to each other directly over a device to device connection.

Device to Device Communication

Direct Device to Device Communication

Personal devices like your home computer or your mobile phone connect to the internet through your Internet Service Provider. In contrast with Internet Web Servers, your personal devices don't have a public internet address that allows other personal devices to connect to them directly. That means that when you use an app to interact with somebody else, you are always going through private servers owned by the company that built that app. This company profits either by charging you a fee for their service, taking a cut on your transactions, forcing you to consume advertisements, mining your personal information, spying on your communication, or raising their valuation for having you on their private network or any combination of the previous.

PROBLEM

Today it's almost impossible to have trustless direct device to device communications

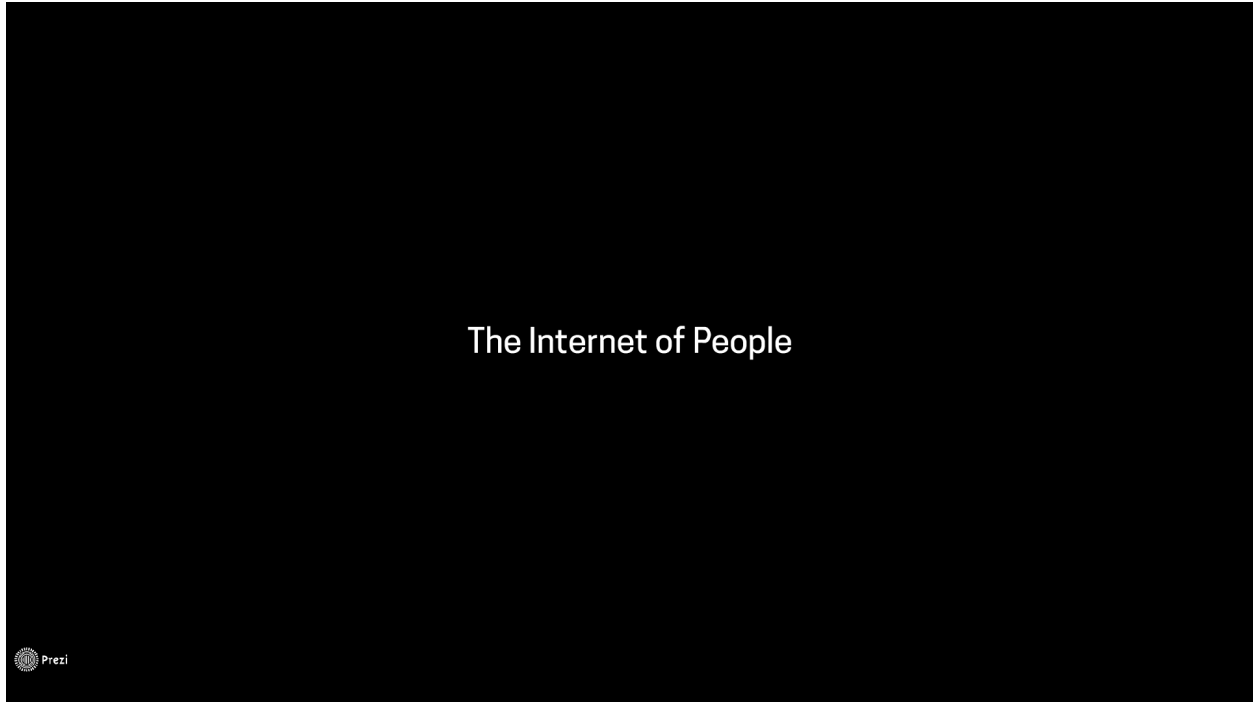
SOLUTION

The Internet of People

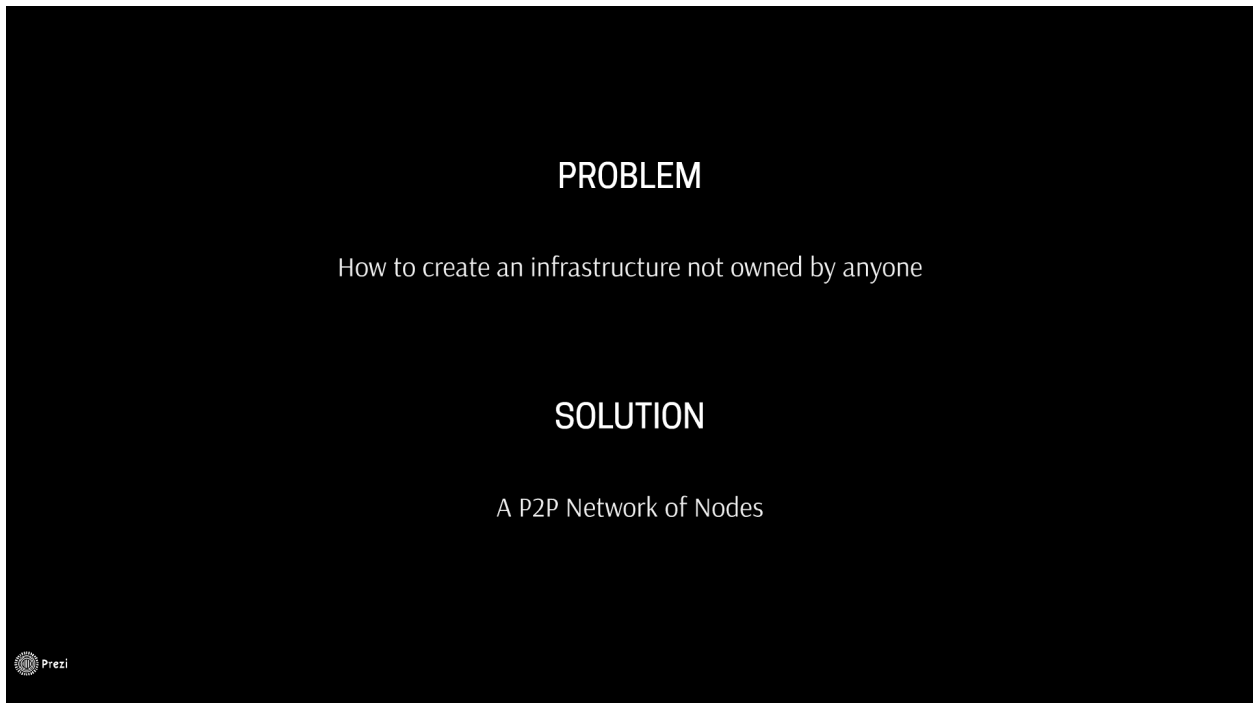


There is no physical limitation for my phone to be connected directly to your phone over the internet. In the end smartphones are also computers. They are unable to do so today because it is against the interest of powerful middlemen that profit from acting as an intermediary between you and me. Artificial barriers are in place today preventing these device to device connections to take place. The current solutions available always involve trusting a third party middlemen to establish that connection and the risk associated with that trust. The good news is that these technical barriers can be overcome without needing to trust any particular third party. We have a plan for that and the infrastructure needed to be built is what we call The Internet of People.

The Internet of People



The Internet of People is the new infrastructure we need to create to enable people to communicate with each other using their own devices directly without relying on any trusted third party to establish the connection. Once this infrastructure is ready and deployed, we will have enabled device to device P2P Apps.



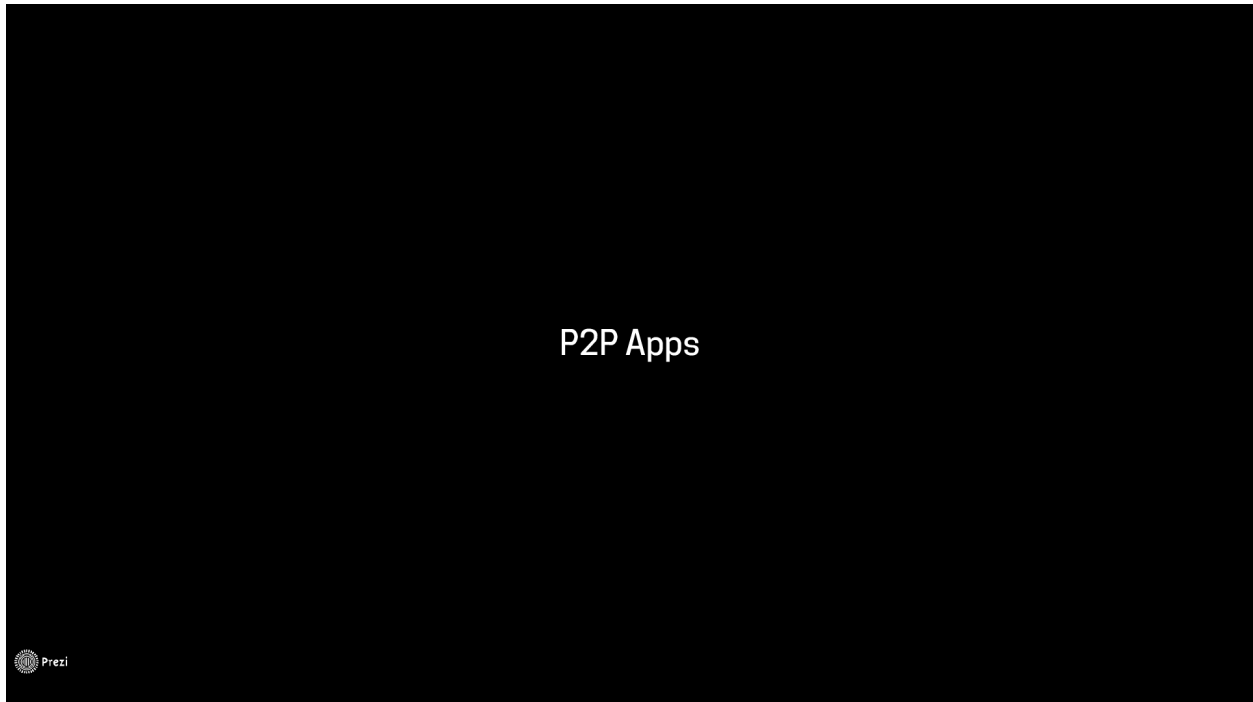
This new infrastructure needs to run on top of the Internet and ought to be as decentralized as the Internet itself. That means that no one will own this infrastructure and no one should be able to control it. The Internet is a network of different networks owned and operated by many different companies and institutions. None of these networks in particular is needed for the Internet to function. We need to inherit the same properties in order to create an infrastructure as resilient as the Internet itself. The way to do it is with a P2P Network of nodes that can be owned and operated privately but none of them is a single point of failure, meaning that the Network itself can not be controlled by any particular entity.

	The Internet	The IoP
Network Of	Networks	Nodes
Main Operators	Internet Service Providers	Node Operators
Business Model	Pay per access subscription	Pay per access subscription
Owner	No one	No one

Prezi

Internet Service Providers, the most common operators of these networks, profit from subscriptions paid by end users to access the Internet. These subscriptions are independent of the activities end users do on the network or the business they run on top of it. Internet of People node operators will also charge end users subscription fees to use their nodes or the IoP Network in general. These subscriptions will also be independent of the purpose of the usage and the business that individuals might run on top of it. The great thing about IoP client software is that it will do everything possible to hide these costs from end users and manage incoming streams of tokens to automatically pay for these subscriptions fees, producing to end users the perception that the service the IoP networks delivers is for free.

P2P Apps (Person to Person Apps)



Since today there are almost no device to device connections, P2P Apps are also not here yet. Once we solve the problem of how to connect one device to another without using trusted third parties to establish the connection, we will enable the existence of directly connected and secure P2P Apps.



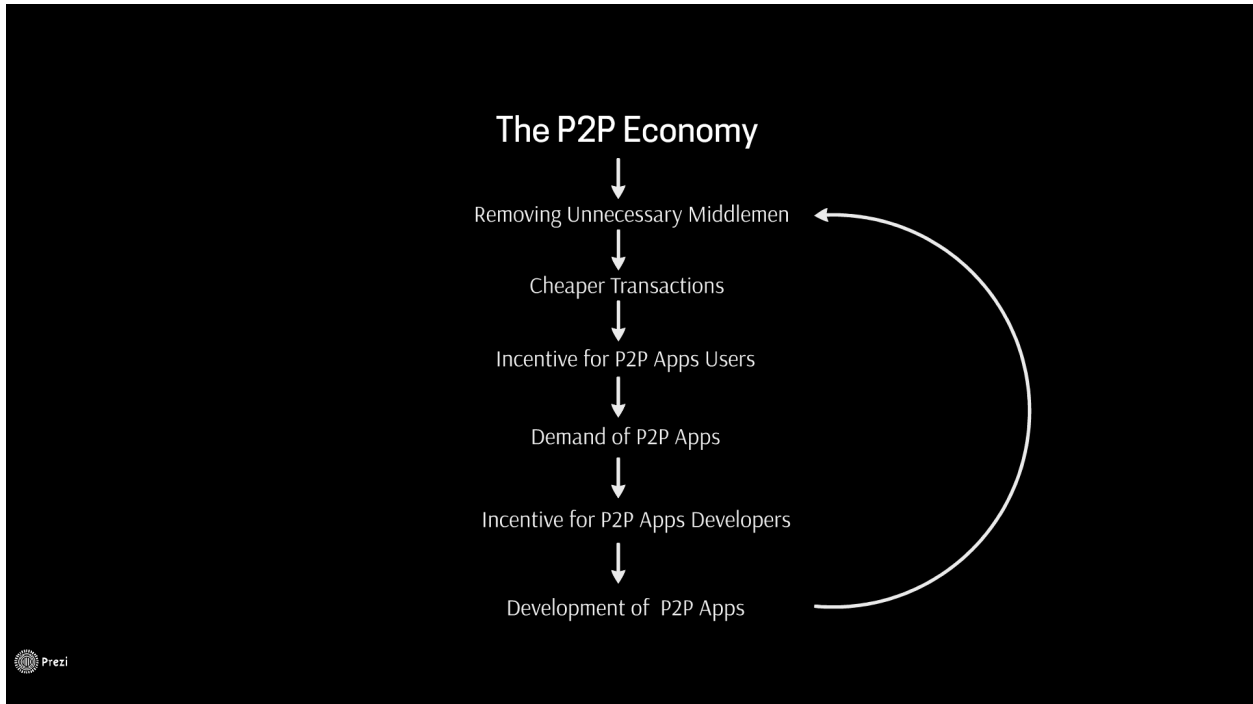
These P2P Apps are never going to be developed if there is no incentive for the developers / entrepreneurs to write their code and market them. At the same time end users of these apps need a strong incentive to use them.

The main incentive for end users to use these apps are cheaper transactions due to extreme disintermediation, or what we call the P2P Economy, as well as increased privacy, anonymity and security. The incentive for P2P Apps developers will come from the demand of new P2P Apps coming from end users already in the P2P Economy.

What is the P2P economy?

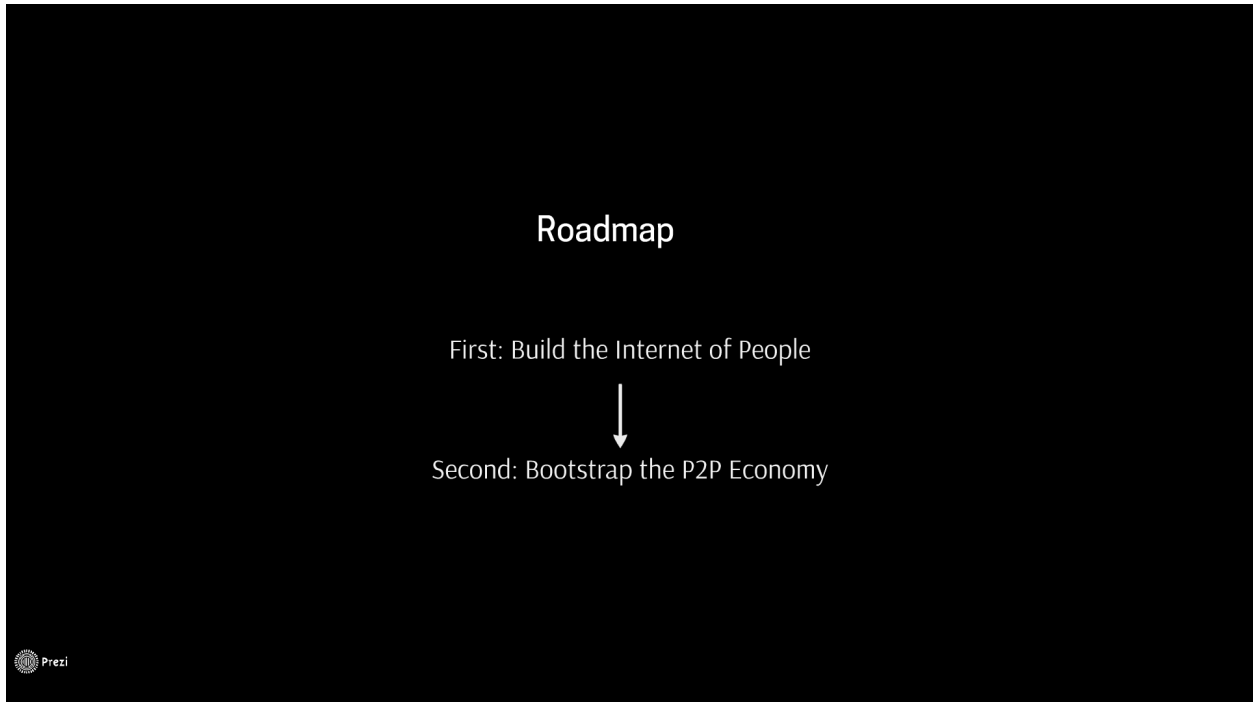


The person to person economy is an area of production, distribution, trade, and consumption of goods and services by different agents in a given geographical location when there is no company as intermediary between consumers who don't know each other or if there is its influence is minimized. In other words is the economy resulting of extreme disintermediation, where most of the middlemen are removed.



Saving money on a consistent basis across several use cases is a powerful motivation for end users to switch from privately operated networks to the P2P economy. We need the P2P Economy because it will become the engine that propels the usage of P2P Apps, which in turn will create a demand for more P2P Apps and thus an incentive for P2P Apps developers. The P2P economy is a virtuous cycle and it stops growing when almost all middlemen are removed and almost all value stays between the parties involved in each transaction. The P2P economy coupled with the IoP has the extra advantage over centralized platforms of permissionless innovation.

RoadMap

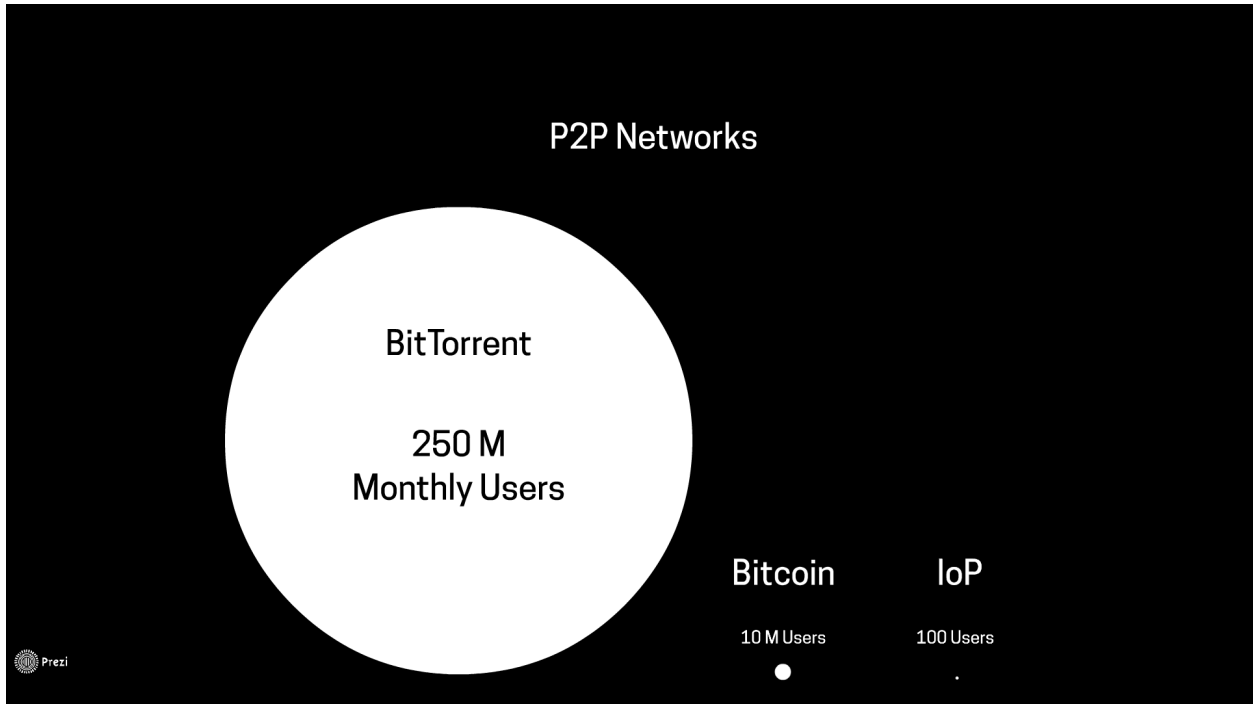


To fulfill the Fermat vision and to bring about a future where people can freely interact electronically without the unwanted interference of redundant third parties, we first need to build the Internet of People and later bootstrap the P2P Economy. Once bootstrapped the P2P Economy will grow by itself spanning every industry until there is no more possible disintermediation in any economic activity.

Phase One: Build the Internet of People



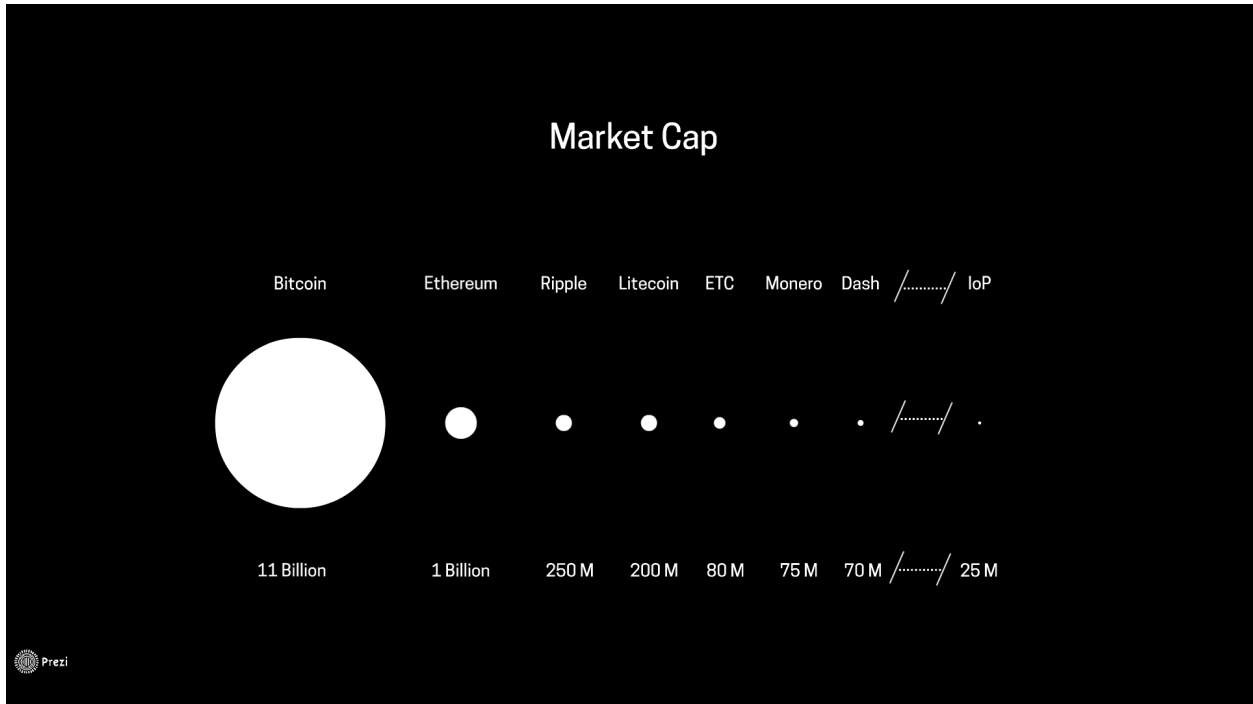
The IoP is a P2P Network whose main purpose is the facilitation of device to device communication. There are several other P2P Networks running on the Internet, each one with different purposes. The largest one is BitTorrent and its purpose is File Sharing. Anyone can run a BitTorrent node and be part of the network. Nodes are servers and clients at the same time, meaning that they retrieve files from other users and they serve files to other users as well. BitTorrent nodes don't need an economic incentive since end users run them to receive files they want for free. Another P2P Network is IPFS, or Inter Planetary File System.



Blockchain technology introduced economically incentivized P2P Networks, meaning that operators can run a node in pursuit of an economic incentive, like transaction fees or mining rewards. The IoP P2P Network is of the same category. This economic incentive is usually received in the network's native token. The IoP network is currently running in beta mode with a tiny amount of nodes, around 30.

Blockchain P2P Networks

Blockchain based P2P Networks usually have client apps consuming services from the network. The bitcoin network for example has several client apps as wallets since the purpose of the Bitcoin network is a digital-cash payment system. IoP client apps can serve any use case, since the IoP network is multi-purpose.



All blockchain networks have a token system that is traded on the market. Thus all of them have a market cap. Bitcoin leads the rankings currently with 11B in market cap. As of November 2016 IoP tokens are to be listed with an expected market cap of 25M.

IoP Network Services



The IoP Network needs to provide several services to client apps in order to achieve its goal of allowing device to device P2P communication without third parties. End users of P2P Apps will need to find each other, agree on having some type of relationship or not (like friends, contacts, etc.), agree on the technical details on how to connect their devices together, and finally connect device to device to conduct business or communicate over that connection.

PROBLEM

How will people find each other?

SOLUTION

Profile Server



Before the IoP Network can establish a connection between my phone and your phone, I need to find you on the Internet and the challenge is to find you without using any trusted third party. That means, no Facebook, no Google, no Whatsapp, without any company involvement at all. Also we extended the challenge to the level where I should be able to find you without you giving me any reference like a phone number and without you sending me any Id or account via unsecure communication channels like chat or email.

Profile Server

The way to go here is to have a Profile Server on the IoP network. Such a server would allow you to upload and host there some searchable information needed to be able to find you. It can be your alias or real name. You could also upload a profile picture and other information you might want to use for people to find you. In this way we expect P2P Apps to allow you to create such profiles, and they would upload them to a Profile Server inside the IoP Network.

PROBLEM

How do I send you a "friend" request?

SOLUTION

Profile Server as Relationship Facilitator



Suppose I found you on the IoP Network. The next thing is to establish a relationship with you at a human level. Your phone posting your profile to a Profile Server is not enough for me to be able to send you a message directly to your device, since chances are you won't have a public Internet address. That means that your device is required keep a connection to the Profile Server of the IoP Network where your profile is being hosted. This will allow me to send a "friend" request using your Profile Server as intermediary. Once we establish a human relationship, I can consider interacting with you using one of my P2P Apps.

PROBLEM

How do I know which P2P Apps you are using?

SOLUTION

Profile Server with Application Services



Let's say for example that I want to chat with you. What I would do then is to enter into my P2P App for chatting. This App will ask your Profile Server which Application Services you have running at your device. There are tons of different Application Services, and P2P chat Apps use one called Chat Application Service. Every time your device gets internet connectivity, the P2P Apps on your device will tell your Profile Server that you are online, and they will send a list of the Application Services available at your device. If Chat is one of them it means that you have at least one P2P Chat app installed. It also means that my P2P Chat app can get ready to talk to your P2P Chat app, because it now knows that you have one.

PROBLEM

How do our devices agree on the technical details of a direct P2P connection between them?

SOLUTION

Profile Server with Signaling Services



To connect one device to another when both are behind firewalls (meaning that they don't have public Internet addresses and also the Internet traffic is filtered by these firewalls) there are a set of technical workarounds called NAT Traversal (these workarounds are performed by special servers called STUN). Most of the time these tricks work and both devices get connected directly one to the other. In a small percentage of times these tricks don't work and the only solution is to use a specialized server to relay the information between devices (these are called TURN servers). The negotiation between devices regarding which STUN server to use, and other technical details is called Signaling Process, and in our case the Profile Server is equipped with this feature.


STUN Server

PROBLEM

Who can help our devices to establish a direct p2p connection between them?

SOLUTION

STUN Server

 Prezi

STUN servers are a known technology so there is no need for us to redo that. Instead we can use an already existing open source implementation and integrate it as part of the IoP Network. The only thing we need to do in this case is to add the mechanisms necessary in order for the operator of this type of server to have an economic incentive.

TURN Server

PROBLEM

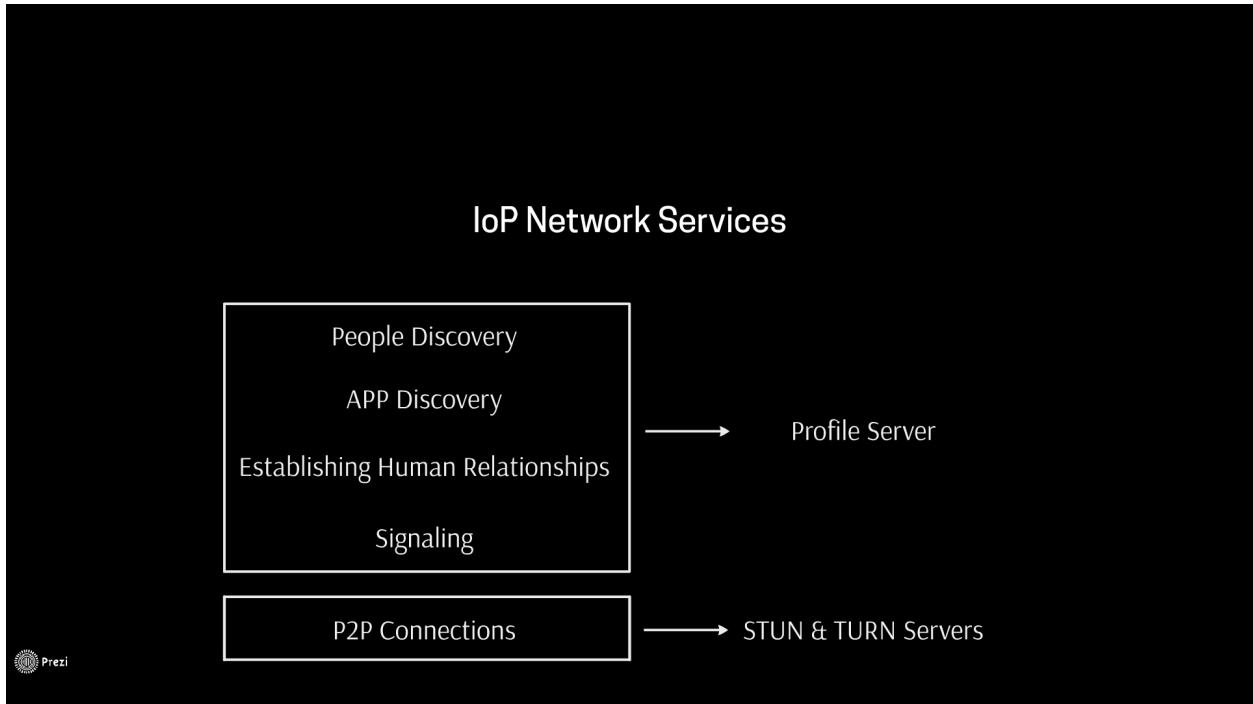
How do we communicate when STUN Servers can't get us connected device to device?

SOLUTION

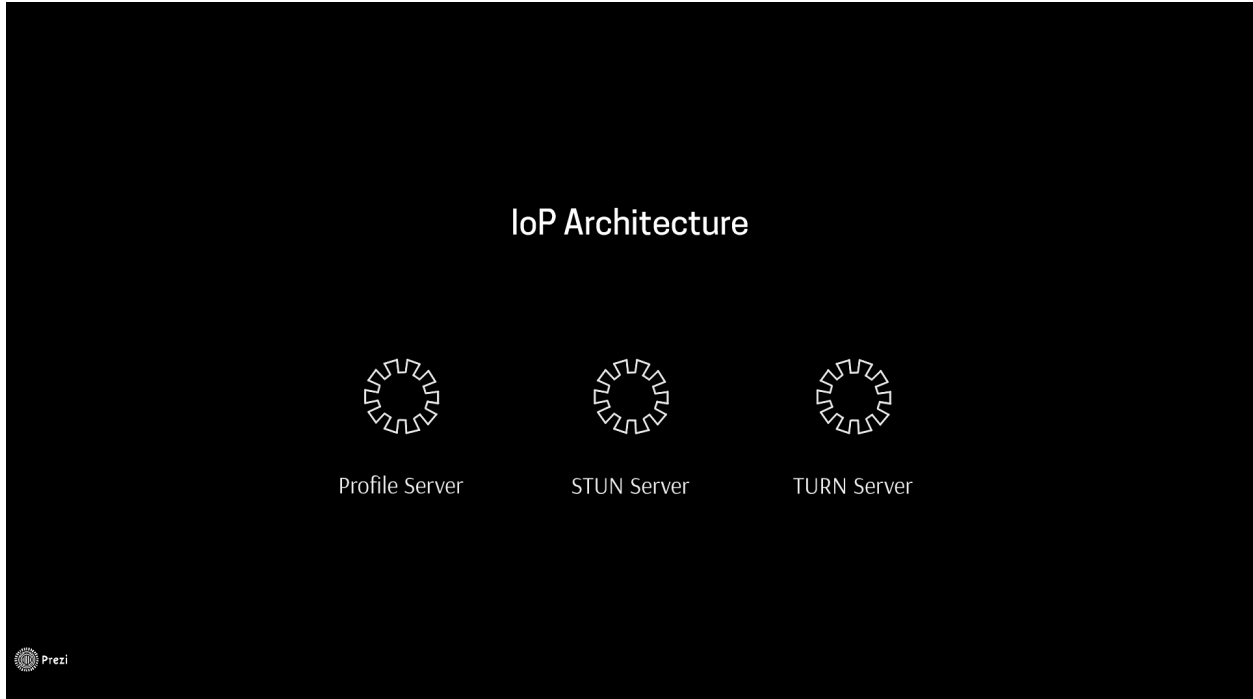
TURN Server



If the STUN server fails to get us connected device to device because some special firewall configurations, the only solution is to communicate via a TURN server. This will forward you all the information I send, and it will forward me all the information you send to me. This technology already exists so the only thing we need to do here is to take one open source implementation and add it to the incentive mechanism for operators.

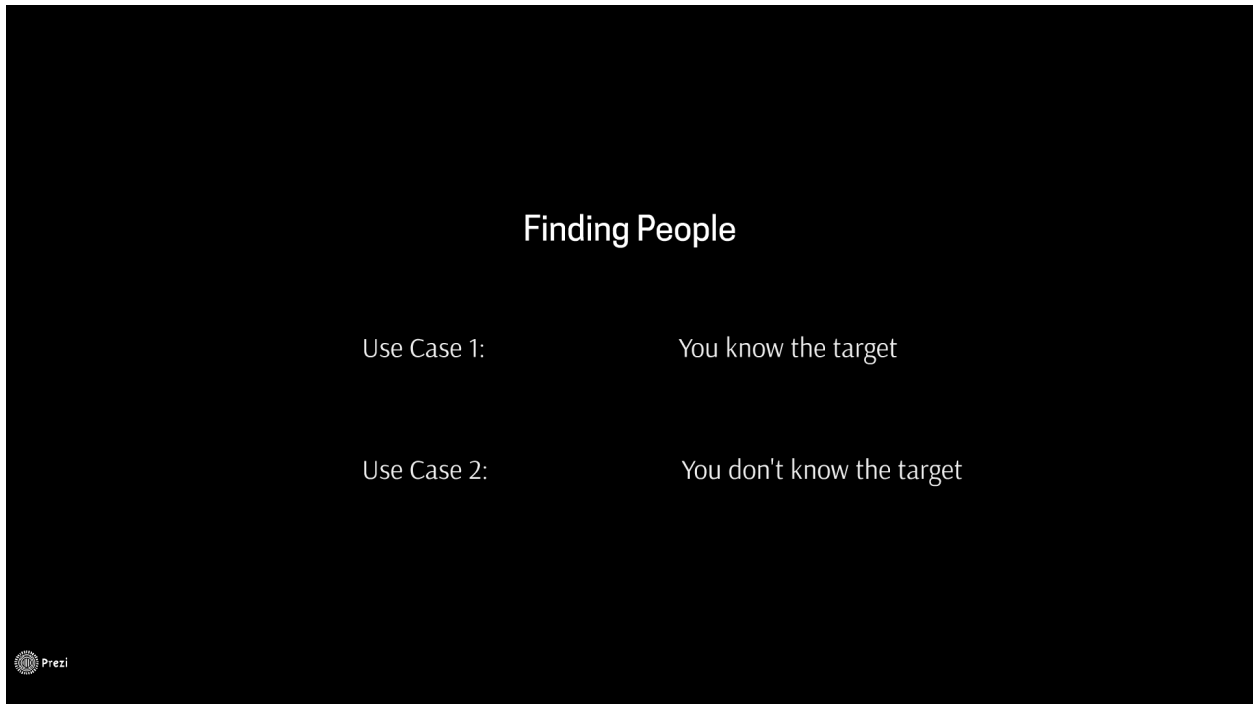


So far we can say that the IoP Network is made of three different types of servers, each one with a small set of responsibilities.



Ok but how does my device selects a Profile Server to host my profiles? How does it find the available servers? We stated before that people's identity shouldn't rely on identity providers like Facebook, Google, LinkedIn, telephone


numbers, etc. There are two different situations in which you would want to find somebody: First you know that person, second you don't know that person.



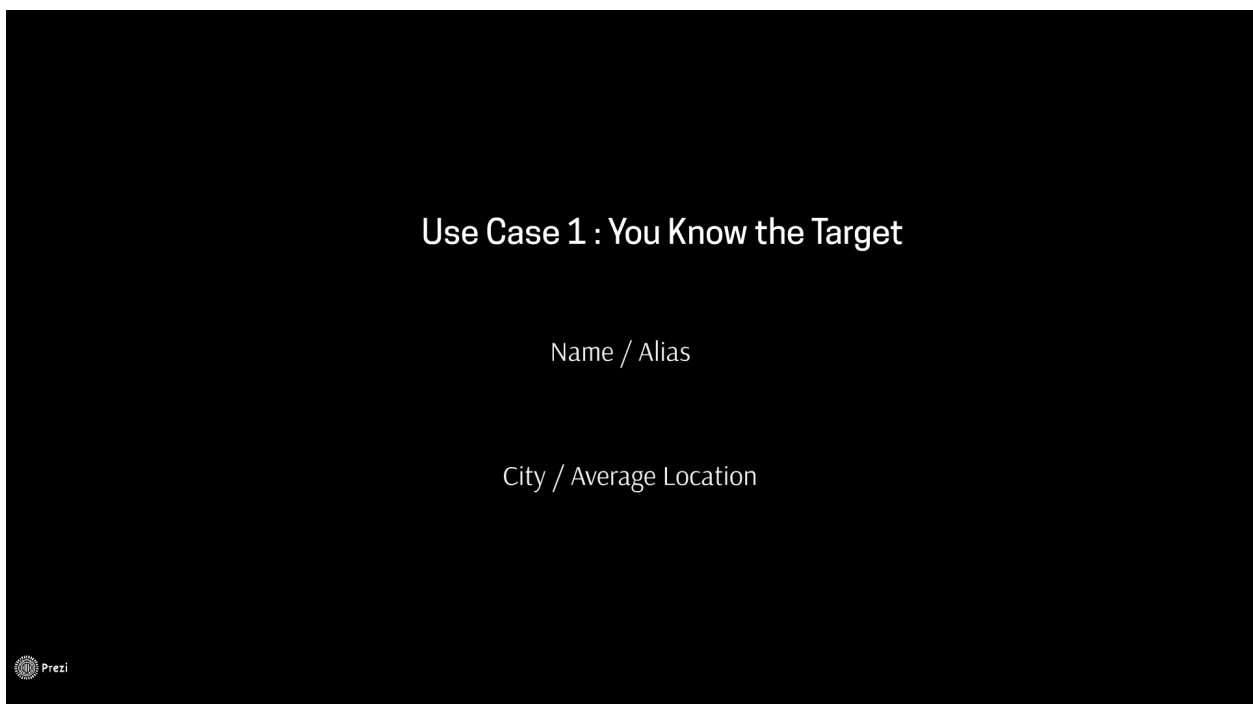
Finding People

Use Case 1: You know the target

Use Case 2: You don't know the target

 Prezi


If you know the target that means that you know her name and usually the city where she lives. A way of finding people without any other reference other than their name and average location implies organizing the profiles by location.



Use Case 1 : You Know the Target

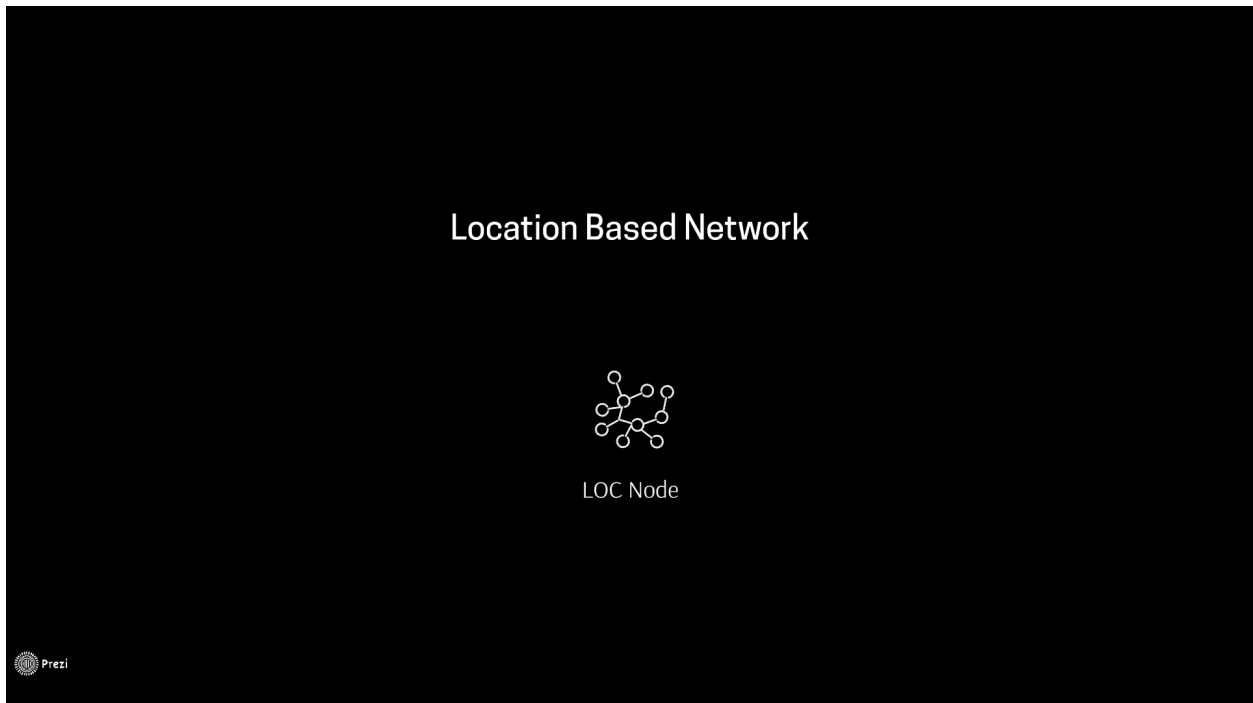
Name / Alias

City / Average Location

 Prezi

With such a minimum amount of information we don't expect efficiency on the search but at least to make it possible, even if that means asking several Profile Servers if they know my friend. But before being able to do that we need P2P Apps to use Profile Servers not too far from the average location of the searched end users. In this way other end users will be able to find them with a minimum set of information. To make that happen we need Profile Servers to run on top of a Location Based P2P Network.

Location Based Network



This P2P network is geo localized. That means that each node defines a physical position on earth (it doesn't need to be very accurate). Once in the network, they will attract P2P Apps willing to host end users profiles in the nearby region. If a P2P App picks a Profile Server in the city of end users of that App, then that would allow anyone in the world that knows those users to find them in the network without any external Id or reference.

Use Case 2 : You Don't Know the Target

Option A : You know their Profile PK

Option B: You know something they do / sell / like , etc.



The second scenario is when you want to find someone but you don't know that person. In this situation there are two possible options: It is a specific person and you know his ID, or you are trying to find someone that is selling something, doing something, interested in something, etc. For the first scenario we need an index of all profiles by their ID. For the second option we need a way to explore the network, ask Profile Servers certain queries and see what they answer, analyze it, continue asking until you find the person you want.

PROBLEM

How do we Index profiles PK in a P2P network?

SOLUTION

Distributed Hash Table (DHT)



In P2P networks, when you need to maintain a distributed index of something across the whole network, the technology used is called a Distributed Hash Table. There are many different open source implementations we could reuse.

PROBLEM

How can we load balance a Profile Server when it is hosting highly demanded profiles from very popular people?

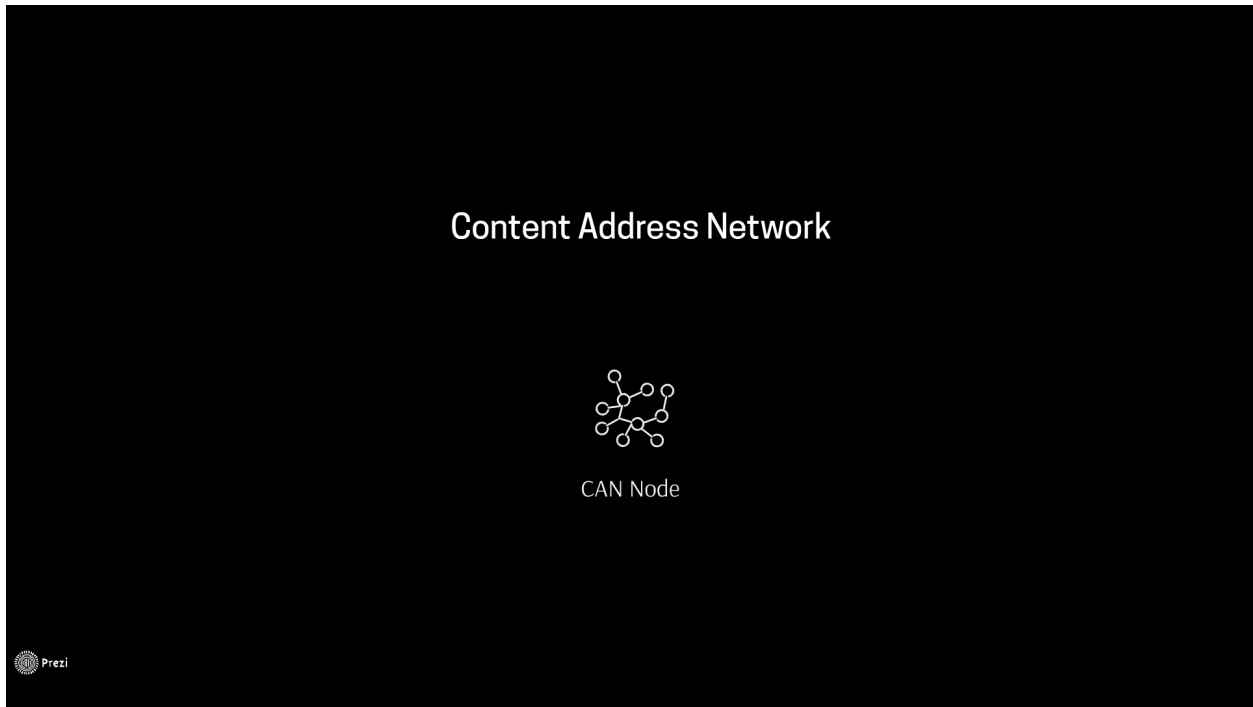
SOLUTION

Content Address Network (Includes a DHT)

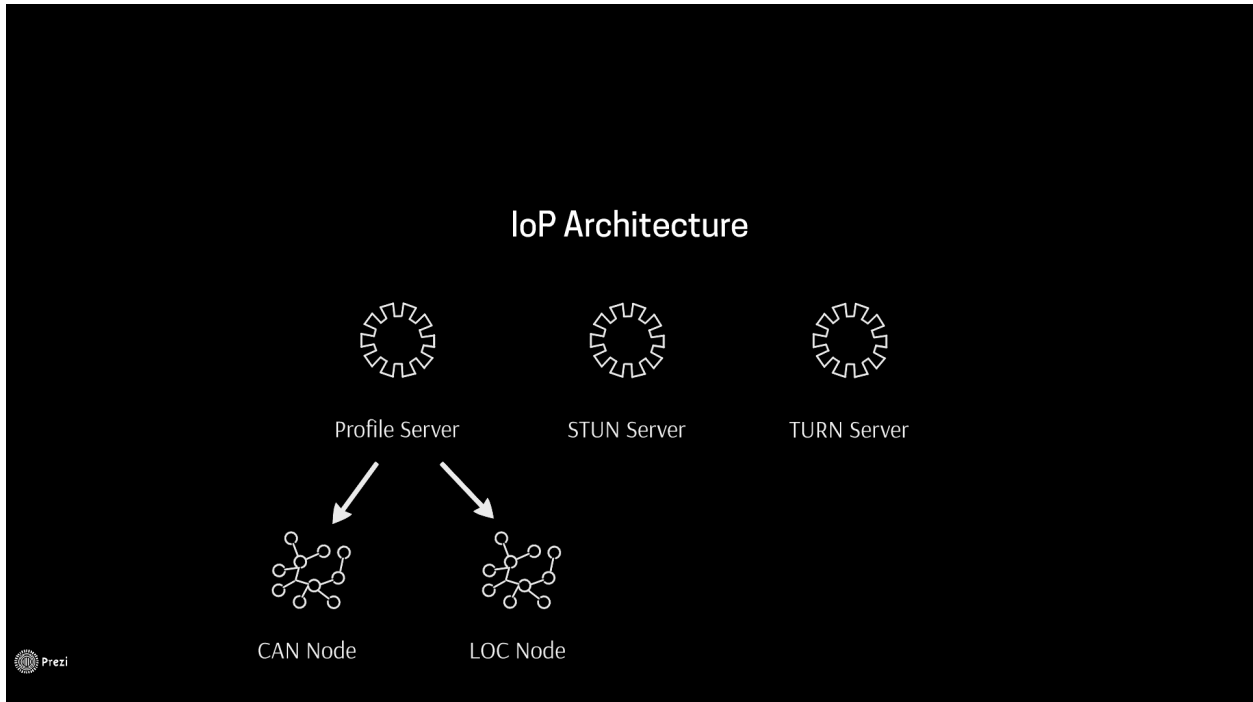


There is another situation related to performance when a single Profile Server is hosting very popular profiles. In this situation the Profile Server can get saturated easily if a lot of people request their information at the same time. There is one solution available that solves both problems: load balancing and the indexing by ID. That is a Content Address Network.

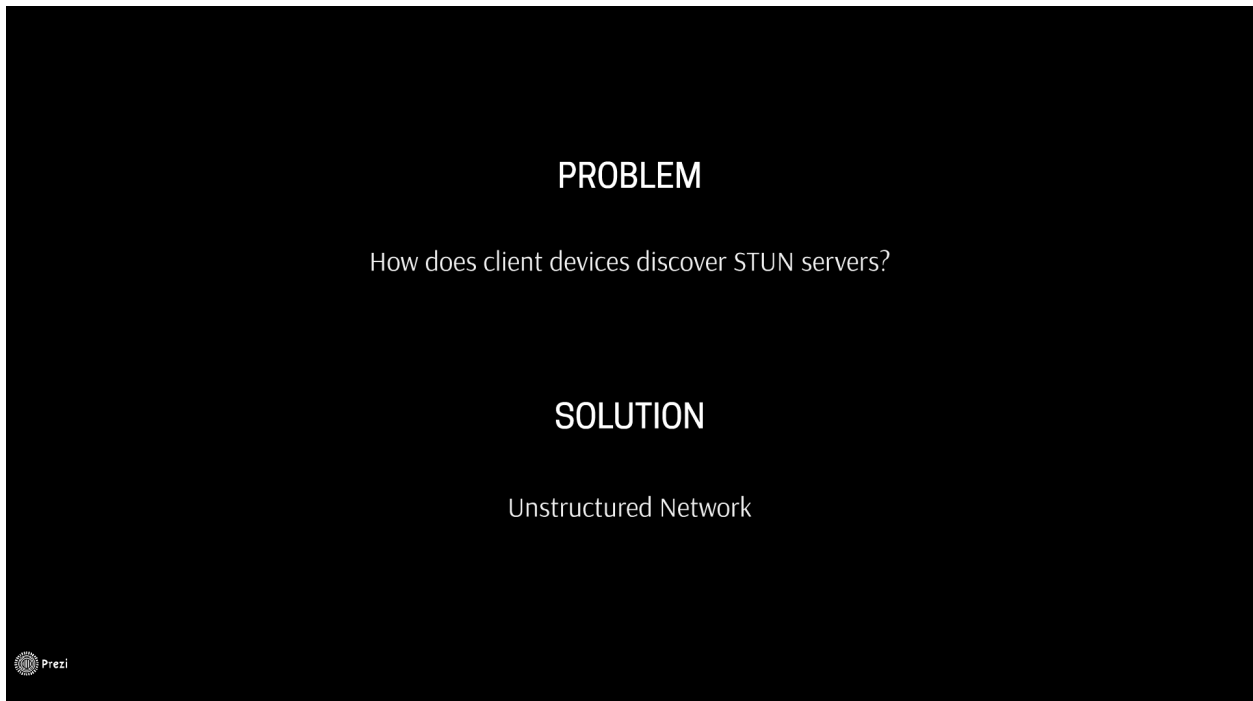
Content Address Network



We chose to use a popular implementation of a Content Address Network and allow Profile Servers to push their profile info into this CAN Network becoming a seeder of those profiles. This enables them to serve profile information by themselves, but also through the CAN, benefiting from its caching capabilities, translated into load balancing of the popular profiles.

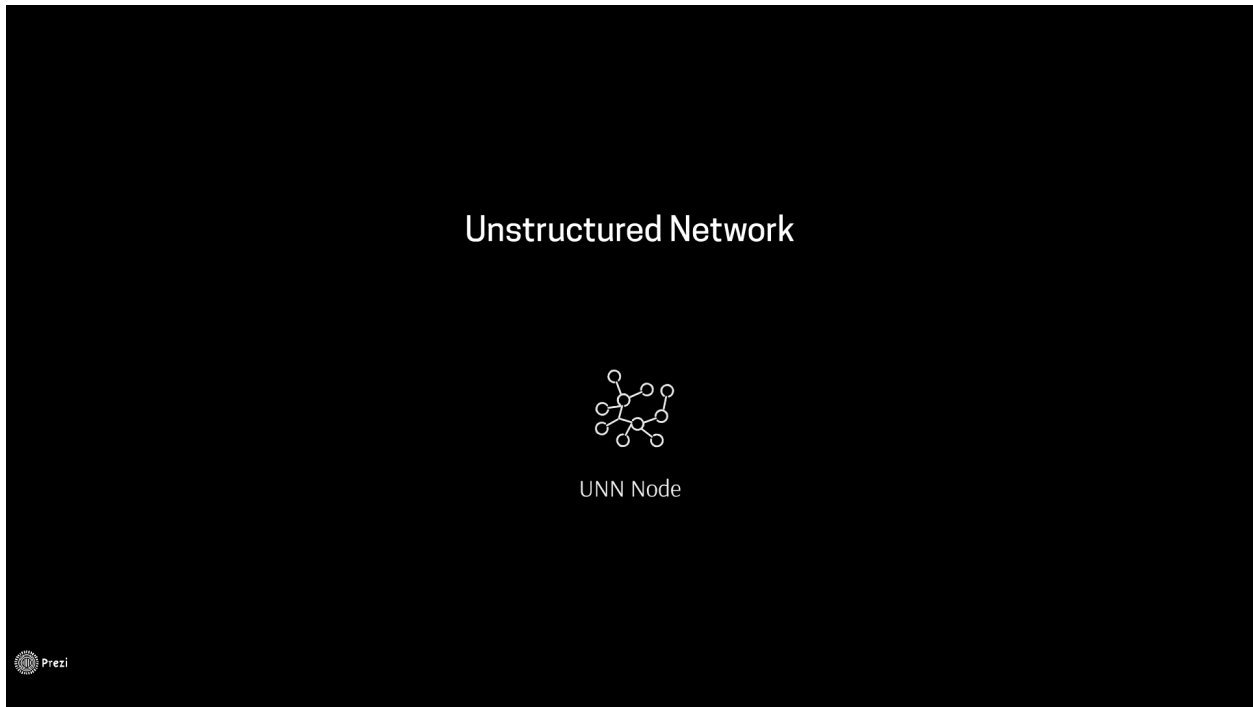


So now we know that Profile Servers works well in standalone mode, but they are also participating in two different P2P Networks: the Content Address Network and the Location Based Network. That raises the question: which network are STUN Servers and TURN Servers running on top of? How do client devices find them?



STUN Servers don't need to be organized by location so they shouldn't run on top of the LOC Network. Their IDs are not known by clients and they don't provide content so they shouldn't run on top of the CAN Network either. The solution is an Unstructured P2P Network.

Unstructured P2P Network



This type of P2P Network as the name says, has no particular structure. Their nodes are randomly interconnected between each other. Usually these networks have a specific purpose or provide a set of specific services. In our case we just need a purposeless unstructured P2P Network, since the service is going to be provided by an external software running on the same hardware. The purpose of the network becomes just to find nodes running together with external service providers of a certain type. In this particular case, STUN servers.

PROBLEM

How does client devices discover TURN servers?

SOLUTION

Latency Based Network



TURN servers provide the service of relaying information between two devices that cannot be directly connected between each other. The nature of this service requires that the latency between these two devices and the TURN server should be minimized. If the two devices could choose a TURN server with the minimum latency then the communication between them would be optimal.

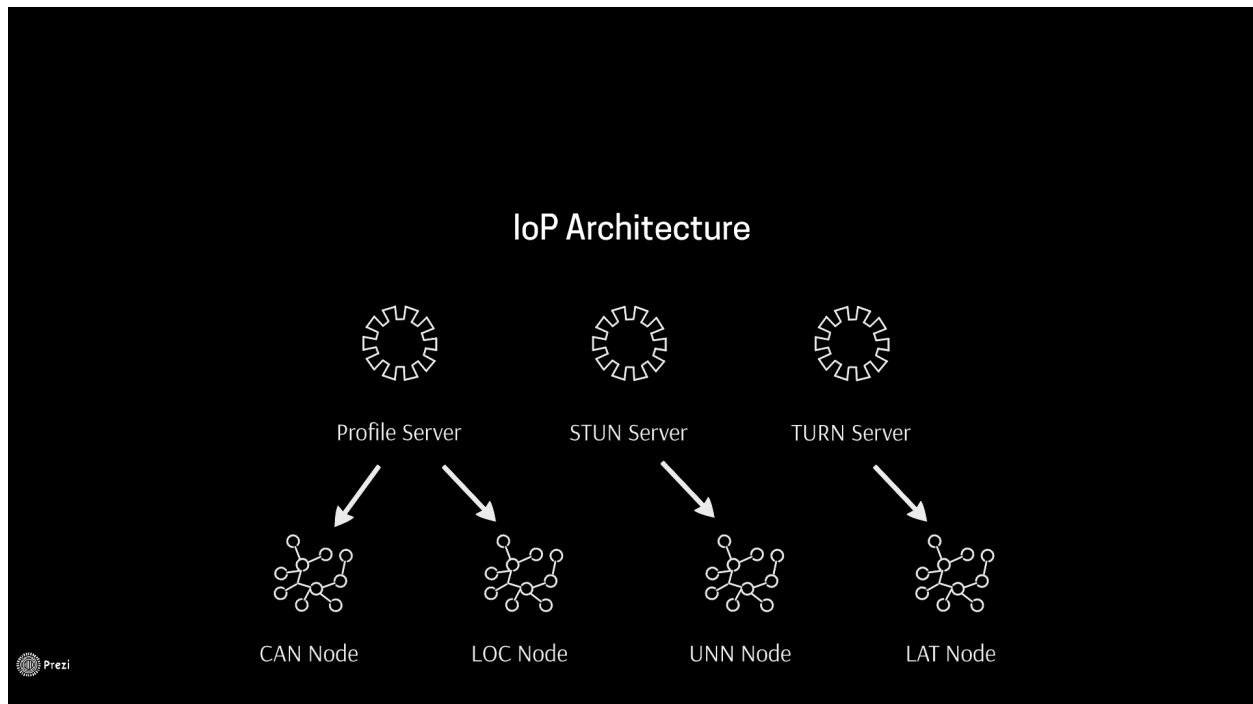
Latency Based Network



LAT Node

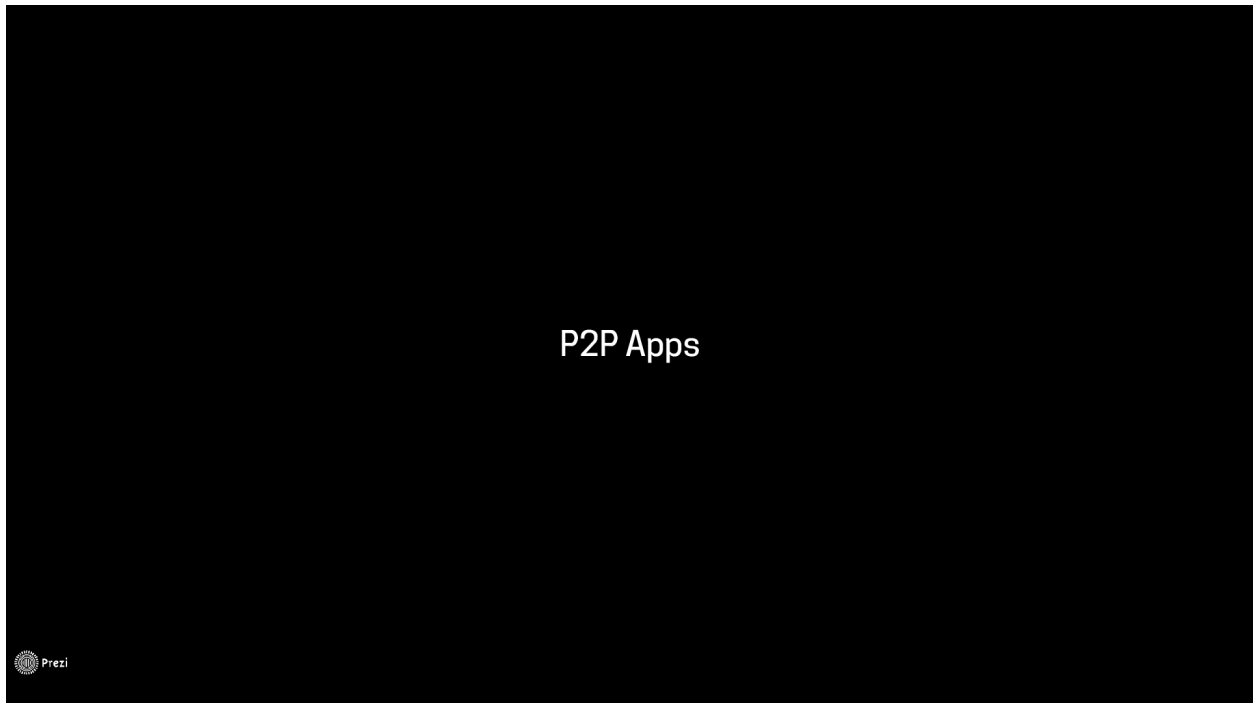


We haven't found so far any implementation of a P2P network organized by latency between their nodes. We have plans to implement our own ideas to solve the previous problem. The objective of this network is to allow TURN servers to run on top of them, and help client devices to pick the best possible TURN server in each situation, based on latency.



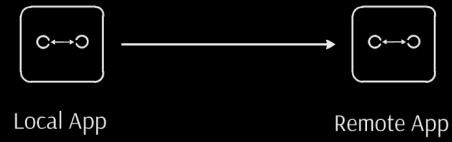
To this point the IoP family of components has grown to 3 different servers and 4 different P2P networks.

P2P Apps



Our concept of P2P Apps is a software that is running at end user devices capable of interacting with an instance of itself running at a remote device or even with an instance of other P2P Apps also running at remote devices. For example, a P2P App could be a chat app. If we provide the means for that chat app to communicate to another instance of itself running at a remote device and both instances can interact, then it falls under the definition of a P2P App.

P2P Apps



P2P Apps Properties

P2P Apps Properties

Local Storage

Local Execution



We envision P2P Apps with the following properties:

1. **Local Data Storage:** The primary and default storage of data is the local device itself.



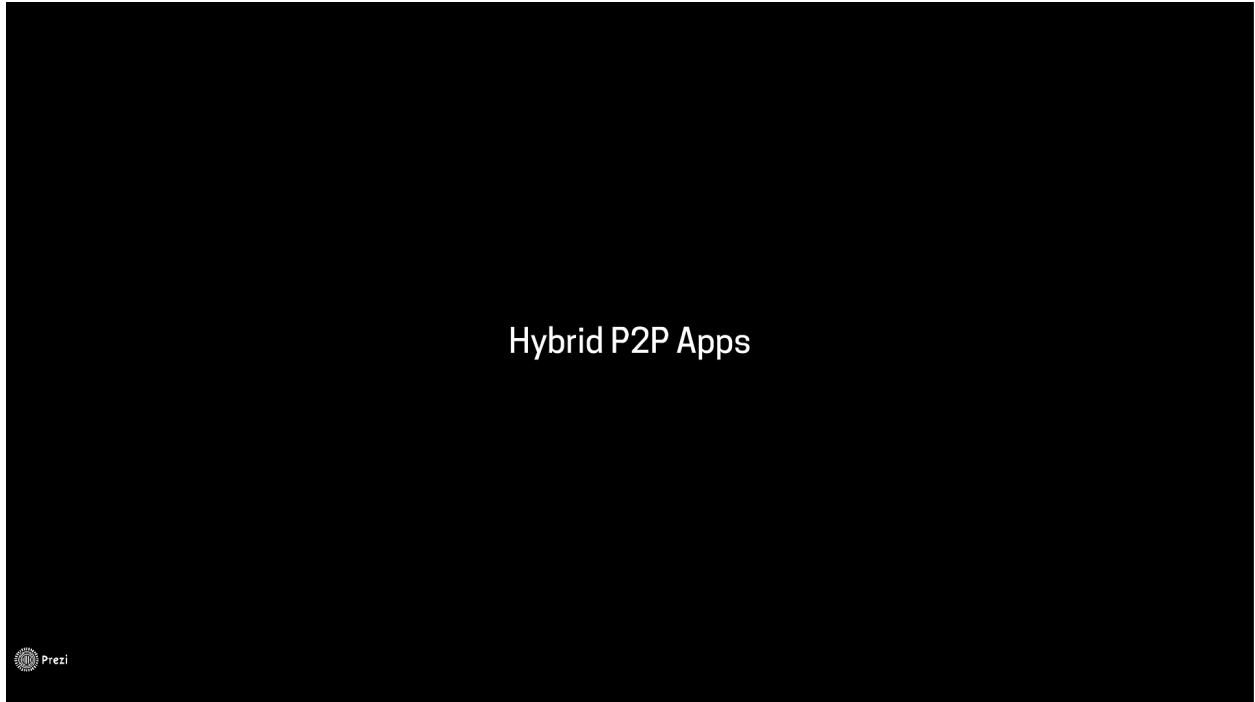
- a. **Index Info:** If there is some information that is needed to locate the end user and connect with him, then this information can be stored outside the local device. End user profiles is a good example. These are stored at Profile Servers.
- b. **Always Online Info:** If there is some information that needs always to be online (we assume end user devices are not), this information might be stored outside end user devices (preferably at a decentralized storage system like Maidsafe, Storj or IPFS).

2. **Local Execution:** The primary and default execution environment for the APPs code is at the local device itself.



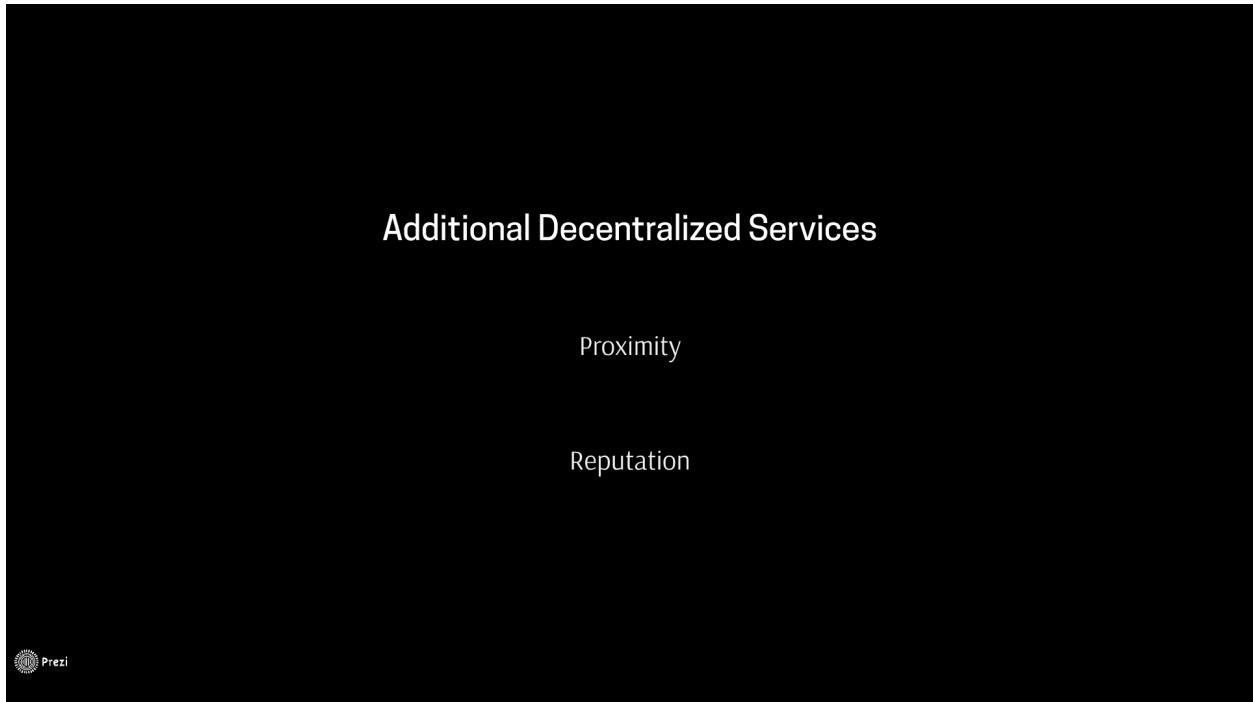
- a. **Multi-Device Execution:** If needed, execution can span more than one device involving several end user devices running the same App. For example, an app running on my phone can request something from an app running on your phone, all this without user intervention. Let's say my app is a cryptocurrency wallet. Once I add you as a contact, my wallet can request an address from your wallet. Your wallet running on your phone can provide that address without your intervention.
- b. **Smart Contracts:** If needed, some part of the logic can be run at a third party execution environment (preferably at decentralized and trustless environments like Ethereum or Rootstock).

Hybrid P2P Apps



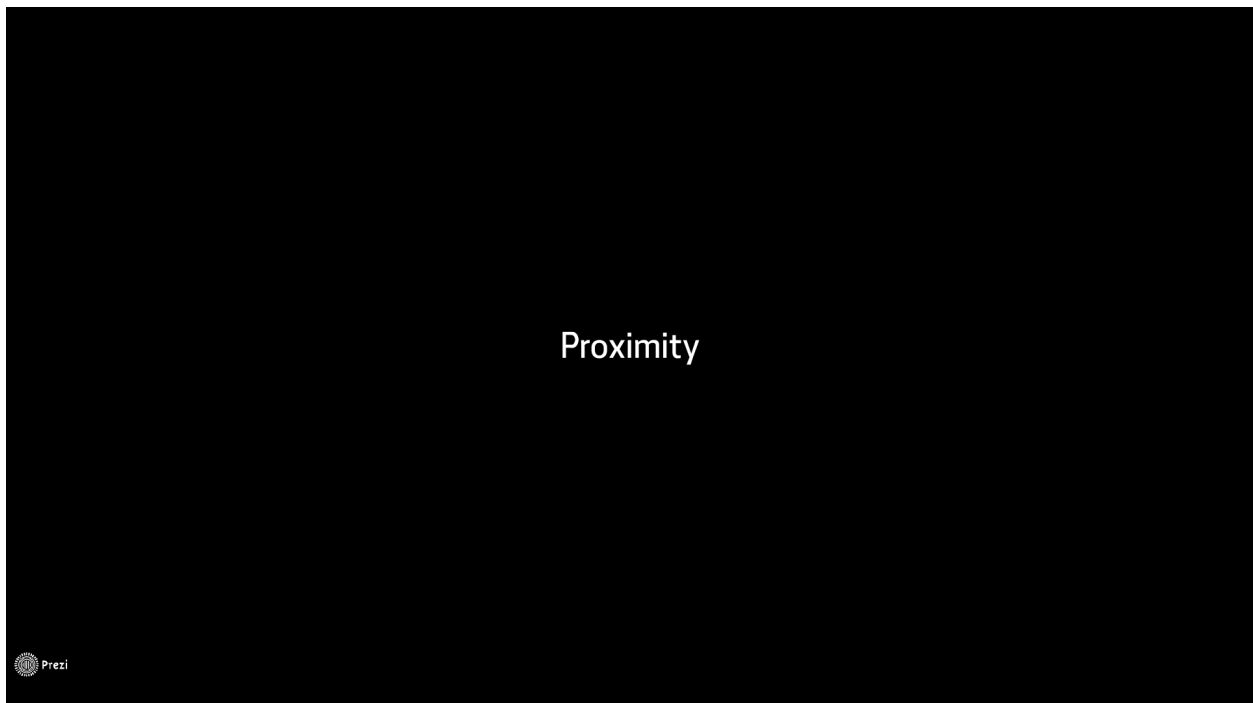
Apps that have these properties but at the same time use some centralized service for storage (like cloud storage) or centralized execution environment (like Amazon Web Services) are called hybrid P2P Apps. By doing so they lose their independence and censorship resistance properties which regular P2P Apps enjoy.

Additional Decentralized Services



P2P Apps running at end users devices solve for a big amount of use cases. However some are impossible to be solved just with P2P Apps. For example Proximity and Reputation.

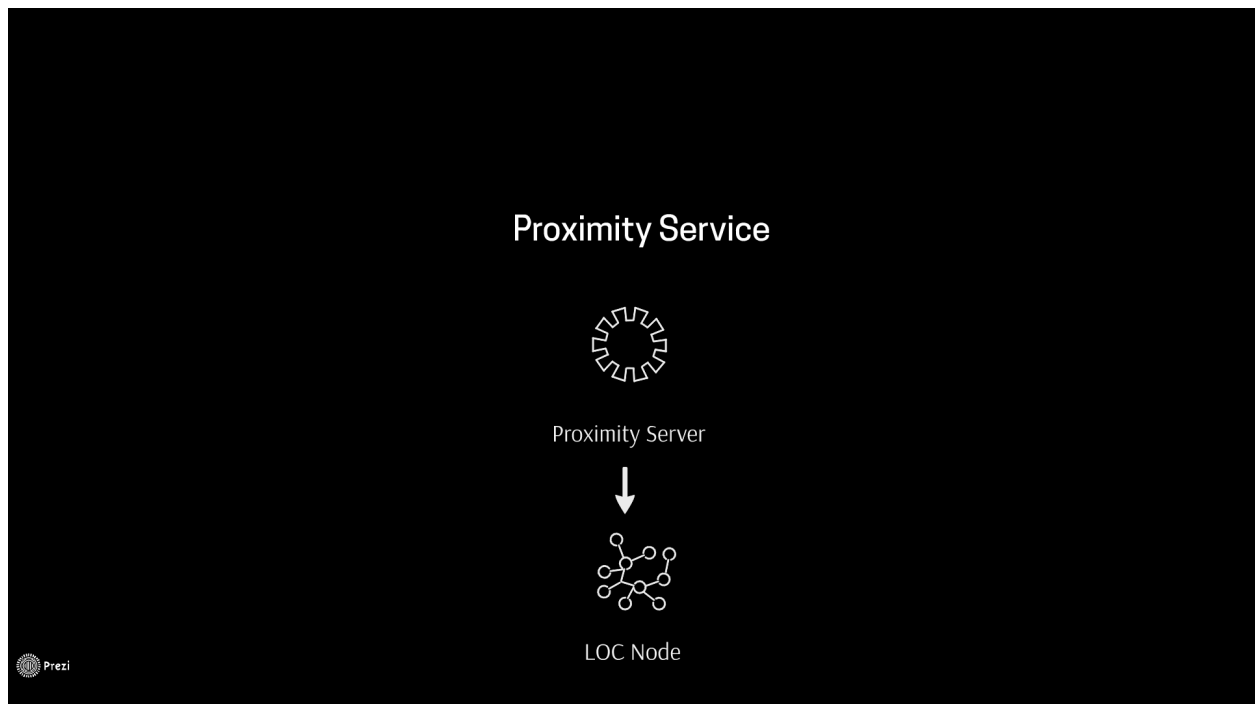
Proximity



There are several Apps that require to find other users by proximity. For example, when the end user of a Taxi Passenger App presses the *Get A Taxi* button, the App needs to find all taxi drivers around that specific user. This App cannot connect randomly to other Apps to see if there is a taxi driver driving nearby.

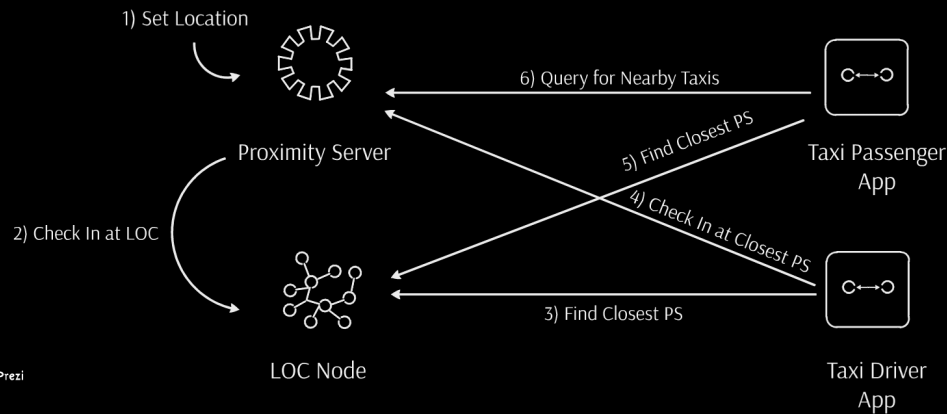
The right way to deal with this situation is letting taxi drivers report where they are and passengers request that information from the same place which is taking the reports. Since we don't want a centralized solution since that would convert P2P Apps into Hybrid P2P Apps, we need a decentralized Proximity Service.

Proximity Service



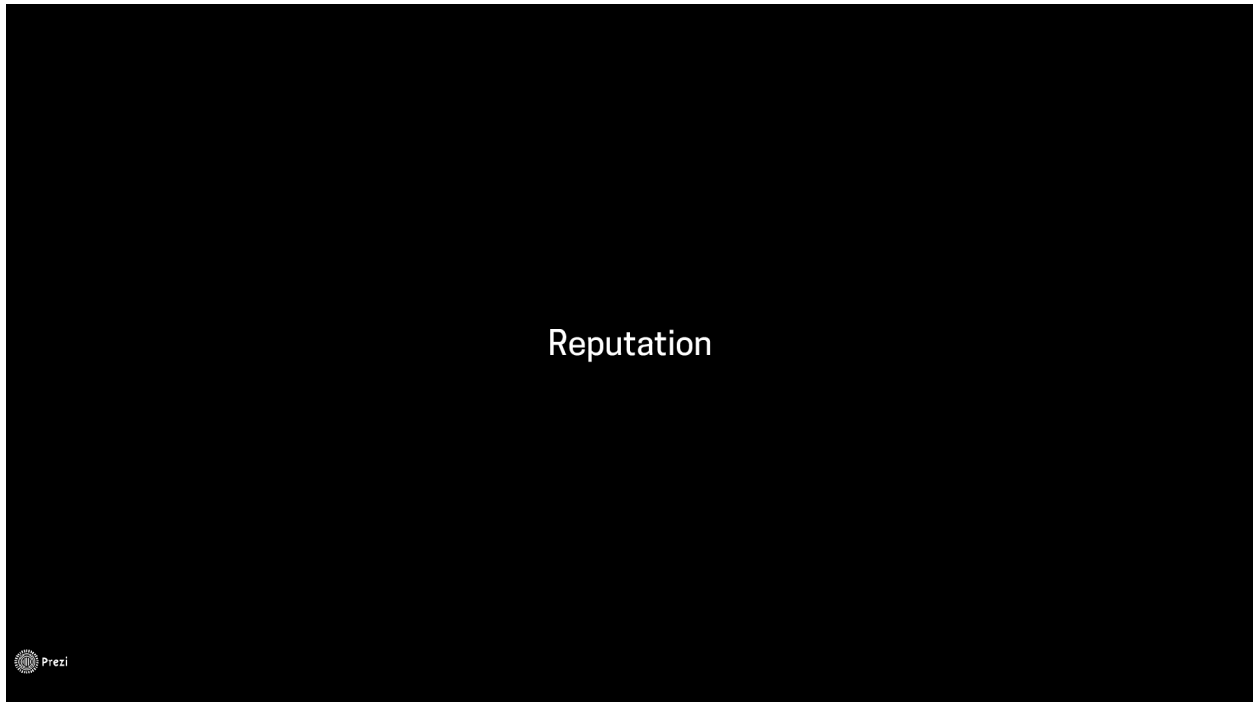
The Proximity Service is an additional service provided by the Internet of People infrastructure. It consists of a geo-located Proximity Server running on top of the Location Based Network (LOC). It works like this:

Proximity Service Use Case



1. Proximity Servers operators initially set a location for their server in terms of latitude and longitude. It doesn't need to be their precise location, but it must be at least at their same city.
2. These Servers check in at the Location Based Network, meaning that anyone can ask this network for the closest Proximity Server to a certain point in the map.
3. Following the previous example, the Taxi Drivers App would find the closest Proximity Server.
4. Then the Taxi Driver App would check in the driver using the App, together with their current position.
5. Taxi passengers also query the LBN to find the Proximity Server closest to them.
6. Then they query this server for taxi drivers driving around.
7. Since taxi drivers are always switching their closest Proximity Server, they are always available to be found by proximity.

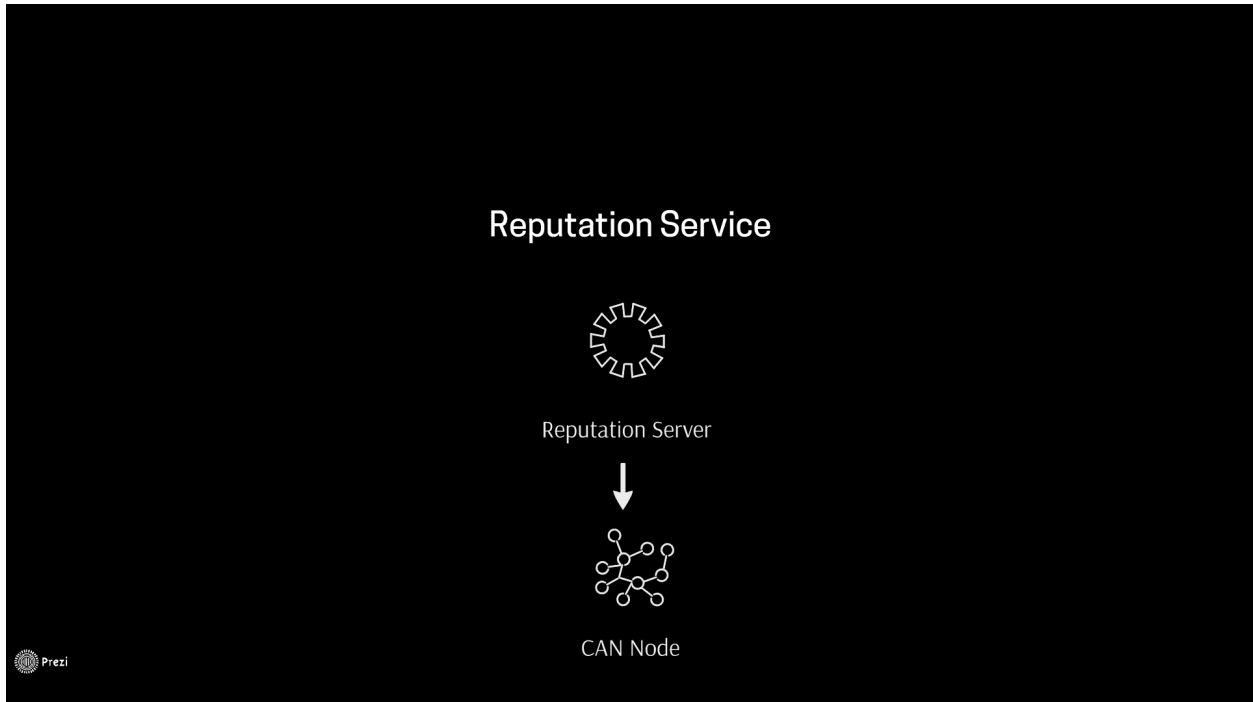
Reputation



People's reputation is a second set of uses cases that can not be resolved just with P2P Apps. It is not possible that end users store their own reputation since that would lead to cheating: they would delete all the bad records in order to look clean. Neither does it make sense that my reputation is stored in your device, since you are not always online, you can not serve that information when it is needed. Moreover, we can not expect P2P Apps to use a centralized solution since that would transform them into hybrid P2P Apps, losing important properties inherent to them.

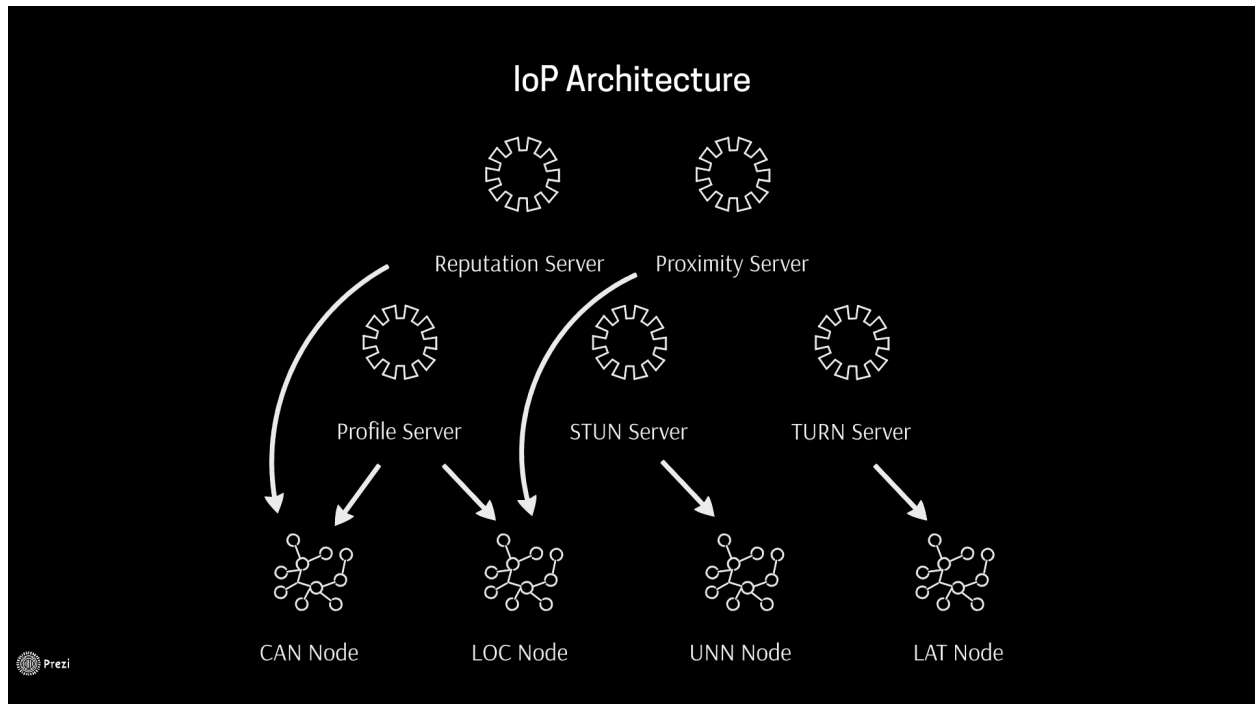
The right way to do it is with a decentralized Reputation Service where anyone can submit reputation information of other people and also retrieve reputation information at anytime.

Reputation Service



The Reputation Service is an additional service provided by the Internet of People. It consists of a Reputation Server that stores and serves reputation information. This server runs on top of the Content Address Network facilitating the access to reputation information across the whole network.

Internet of People Architecture



With the latest additions, the IoP's architecture is expanded to include these new necessary services: Proximity and Reputation.

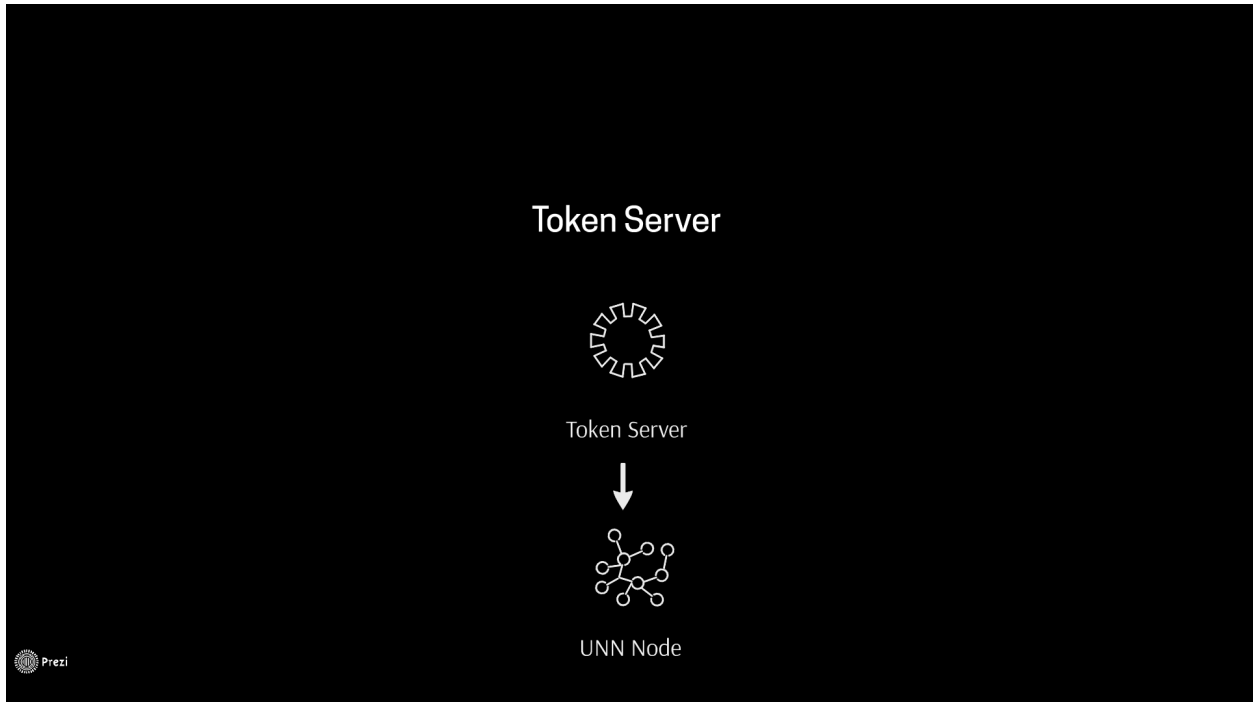
Incentives



Running all these servers and network nodes consumes resources: electricity, disk space, CPU, memory, bandwidth, etc. Each of the servers requires a different set of resources. How will we incentivize node and server operators to run this software?

We need a token system for that. The tokens should have a market value which operators could use to cash out and get the fiat currency or cryptocurrency necessary to pay their expenses and have a net profit.

Token Server



This Server internally has a blockchain that serves as the underlying technology to keep a distributed ledger of token transactions. A bitcoin based blockchain is designed as an electronic cash payment system. This is perfect for us to deploy an application token system that can be used for transactions between software components.

The Token Servers run on top of the Unstructured Network Node, allowing for interaction between Token Servers and between clients and servers.

Incentive Types



Blockchains like the one used in bitcoin, provide two types of incentives: issuing new tokens to some participants of the network, and transaction fees paid by entities transferring value from one account to the other.

The table is titled 'Bitcoin' and is presented on a black background with white text. It has three columns: 'Before', 'Now', and 'Consequences'. The 'Before' column lists: '1 Software Component', 'Both Economic Incentive to Node Operators', and 'Power Distributed across Node Operators'. The 'Now' column lists: 'Many Software Component', 'No Economic Incentive to Nodes Operators', and 'Power Centralized across Mining Farms and Mining Pools Operators'. The 'Consequences' column lists: 'Nodes', 'Less Full Nodes', 'Less Resilience', and 'Power Concentration'. A small circular logo with the word 'Prezi' is in the bottom left corner.

Before	Now	Consequences
1 Software Component	Many Software Component	Nodes
Both Economic Incentive to Node Operators	No Economic Incentive to Nodes Operators	Less Full Nodes
Power Distributed across Node Operators	Power Centralized across Mining Farms and Mining Pools Operators	Less Resilience
		Power Concentration

Bitcoin is designed as a single network and the software does all these tasks in only one component: it processes transactions, mint new tokens and run an unstructured network. So in their case, the two incentives go to whoever runs this software component. This was true for a while, until someone removed the mining operation from that component and ran it outside, enabling the creation of mining farms, and mining pools. Now running a bitcoin software component doesn't have an economic incentive, resulting in a reduction of the number of nodes in the bitcoin network, and a concentration of the mining in a few entities. A reduced number of nodes leads to a less resilient network.

PROBLEM

How do we prevent the centralization of mining?

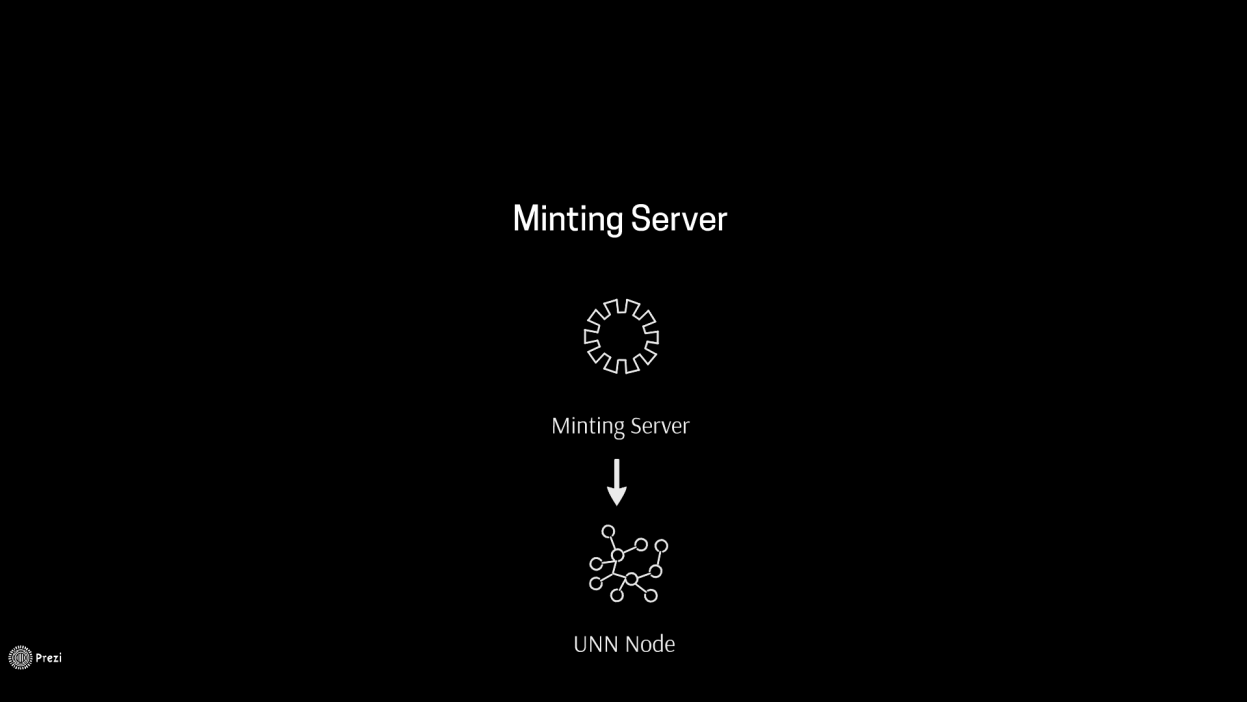
SOLUTION

Minting Server

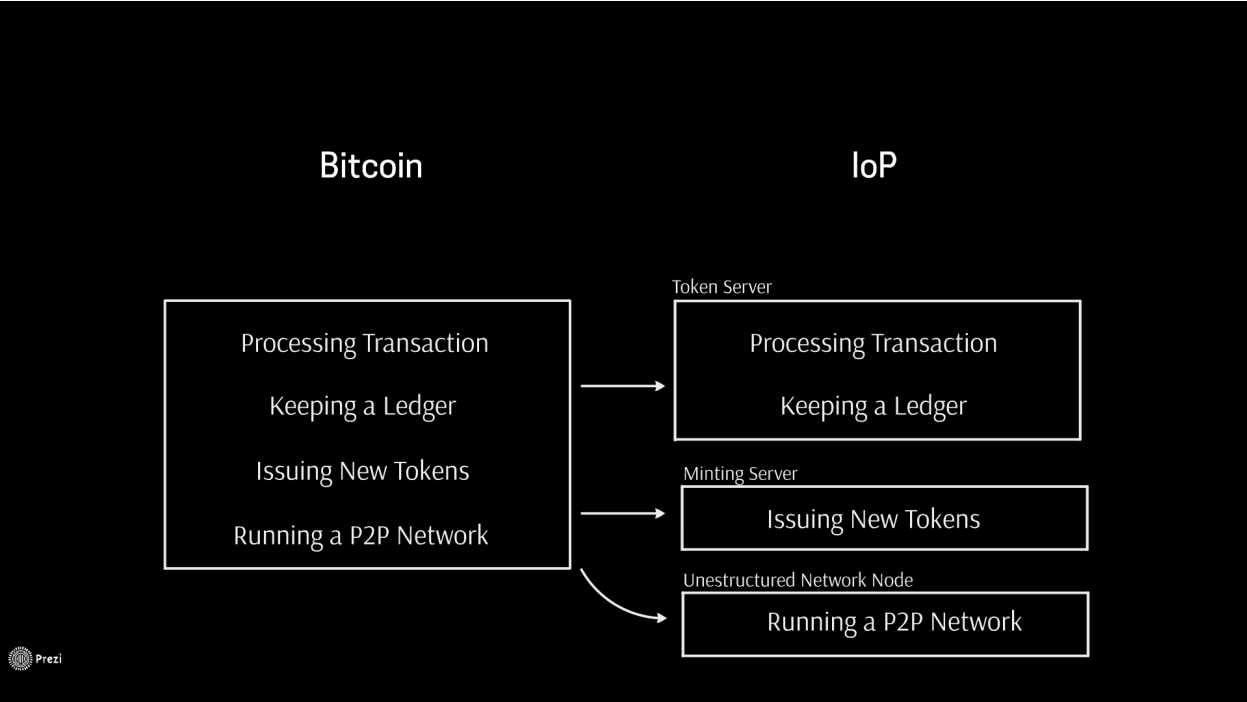
Prezi

How do we prevent the same situation to happen in our IoP networks? The solution is to reward operators with newly issued coins. This is true for operators running the full set of nodes (full nodes) or multiple nodes if running a full node is not possible for them (they might not have public internet addresses to be able to run STUN or TURN servers for example). For that we need Minting Servers.

Minting Server

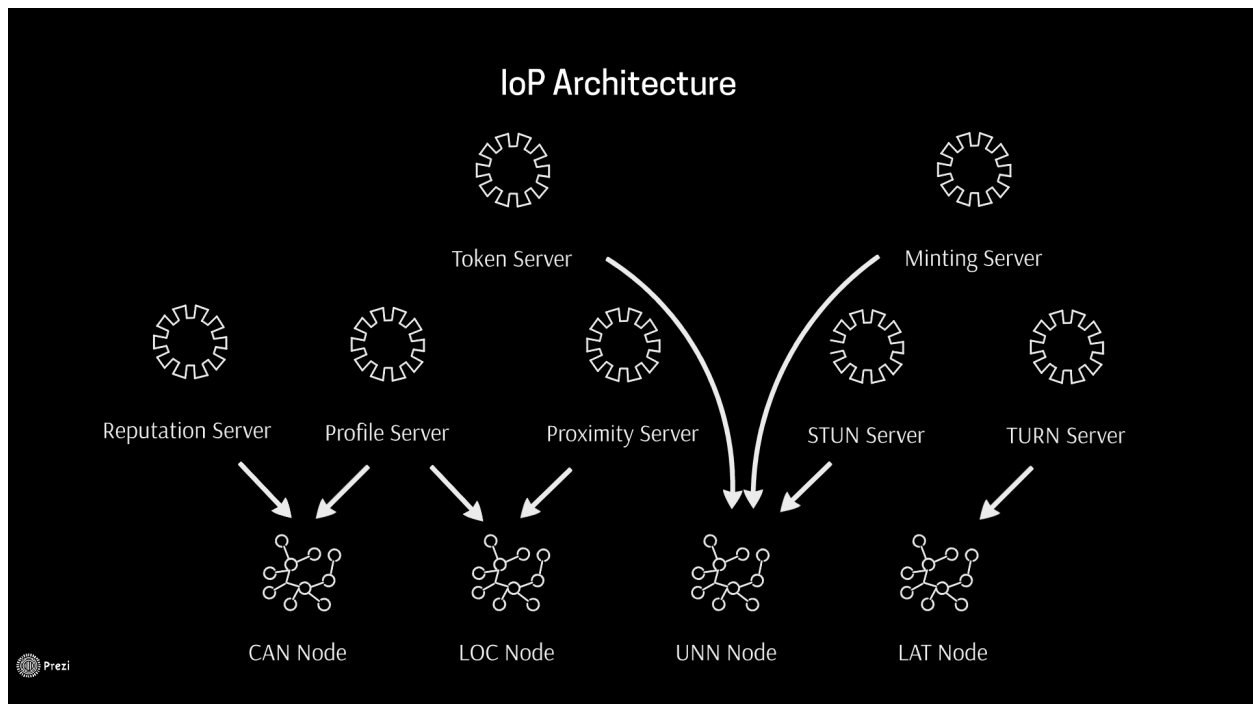


A Minting Server is an IoP software component that runs the logic needed to mint new tokens. As we want to reward full node operators then it must also have the logic to detect who is running full nodes. Minting Servers run on top of an Unstructured Network Node.



The original bitcoin software had multiple functionalities at the same time including minting (or mining). We move that logic to a specialized server in order to allow us to freely add improvements to it without interfering with the transaction processing task or keeping the ledger. We can also see how we diverge from the original bitcoin software, partitioning it in three different IoP components.

Architecture



So far, the IoP Architecture with all its components looks like this.

Conclusion

We have a clear vision of a decentralized human species and we are executing a long term plan to get there. The plan is based on logic, is viable, and makes sense. The Internet of People and the Person to Person Economy are huge innovations necessary to get there. At Fermat we are all very excited to be building today the world we will all live in.