Comparison between the blockchain and ID ecosystem

The purpose of this document is to give an objective comparison between blockchain-related technologies and the ID ecosystem. We compare and contrast the differences and similarities between the two technologies, specifically:

- Blockchain vs ID ecosystem
- Crypto wallet vs ID
- Crypto assets vs assets
- Crypto smart contracts vs contracts

We hope this document will help the reader gain a deeper understanding of the mechanics behind the ID ecosystem.

Note: When we refer to blockchain, we are mainly referring to Bitcoin and Ethereum-based technologies, although many of the comments apply to Blockchain in general.

Blockchain vs ID ecosystem

	Blockchain	ID Ecosystem
Interoperability	Different blockchains have different algorithms and protocols. A token on one network doesn't work on the others. For example, to use Bitcoin on Ethereum, you need Wrapped Bitcoin. To use ETH on Polygon, you need WETH.	All ID clouds have the same structure. The user experience is coherent wherever users go. For example, if a user uses IDZ as the ID provider and switches to another provider later, s/he will continue to be able to use the same assets as if nothing changed.
Speed	Transactions are not instant with blockchain as they need to be verified, included in new blocks and distributed to other blockchain nodes.	Transactions are instant as there is no need to incorporate them into blocks and perform mining.
User experience	Given that the different networks are not interoperable, the user experience is affected.	User experience is always coherent not reliant on the ID provider the user uses.
	Given that the network speed is dependent on the transaction volume, throughput fluctuates according to the network status.	Performance is always consistent as the system can scale dynamically in line with demand.
Scalability	Blockchain is not scalable. Data is only getting heavier and even with faster computers and processors, blockchain will never manage to reach the desired balance between scalability, speed and security. Centralised off-chain solutions become a necessity for the network to be functional.	Given that the ID ecosystem is cloud-based and interoperable, it is scalable. For example, any new provider can scale the same open-source architecture without affecting the user experience.
Security	Transactions are secure. Security keys are in the user's wallet.	Transactions are secure. Security keys are with the user.
	Users can enhance the security by using a hardware or a cold wallet.	Users can enhance the security by buying an ID enabled device.
	No measure of transaction or logging in security. When a transaction is signed by a wallet, there is no record of the type of wallet used (whether it is browser-based, app, hardware, etc.). For example, a hacker can access someone's wallet by just knowing the seed phrase of the wallet.	ZINDEX (or security index) enables users to set different levels of security from low to extreme for executing transactions and ID security. For example, a hacker wouldn't be able to access someone's ID by just knowing the seed phrase. Users case configure additional security layers (e.g., 01 watch, confirmation of identity on an existing device, etc.) to be able to use the ID.
	Some wallets operate in browsers as plug-ins. This imposes security risks, as browsers don't typically contain technology to allow the secure storage of user-sensitive data.	IDs operate as standalone apps.

	Blockchain	ID Ecosystem
Privacy	Transactions are public by nature. For example, users can check each other's balances and assets.	Transactions are private by nature. For example, users can't check each other's balances and assets.
Cost structure	Gas fee to miners, i.e., cost per transaction. The fees could add up to high figures depending on the user's activity during busy times.	Subscription fee to ID cloud provider, i.e., all in one fee. No gas fees.
	Different services can have additional subscription tokens, nonetheless, gas fees are unavoidable.	Different services can have additional subscription assets. No gas fees payable. IDZ envisages that all products and services will become subscription-based in the near future, all manageable from within one's ID.
Environmental impact	Blockchains that utilise proof of work algorithms are notoriously resource-intensive and require an ever-increasing amount of computing power. This has a detrimental impact on the environment.	ID ecosystem transactions are environmentally efficient because there is no need to perform CPU-intensive work to verify transactions.
Data storage	Blockchains (not wallets) store links/pointers to the data that is stored in different off-chain data storage solutions (such as IPFS).	IDs store the actual data itself.
	While pointers to the data are difficult to find, they are made public on the blockchain.	Third parties (not the users) have pointers to the user's data which never leaves his/her ID.
	IPFS networks break the content to "small parts" and store it in "un-encrypted". If any of the IPFS servers that hold any part of the data goes offline, and there are no copies of the data (unless complex IPFS-clusters are utilised), users will not be able to access their content.	The content is stored encrypted as one piece (i.e., not broken to small parts).
Ownership definition	Blockchain networks define ownership according to the following principles: Storage: the data sits in storage network (e.g., IPFS), not with the user. Users own pointers to access the data. Users can share these pointers with third parties.	IDZ defines ownership according to the following principles: Storage: the data sits on the user's device or a cloud space fully controlled by the user. Users can give third parties access to the data.
	Encryption: user assets are not typically encrypted and instead stored "off-chain".	Encryption: asset encryption with unique keys for the whole ecosystem. The data always sits encrypted with the user as one piece, not elsewhere. Users can share access to the data with third parties. For example, in Exchange App, the app only stores pointers to the data which never leaves the ID without the user's consent. Refer to the Exchange use-case: https://idz.com/use-cases/exchange
	Keys with the user.	Keys with the user.
	Open source.	Open source.
	Always immutable.	Immutability is optional (depending on access rights). Having the immutability optional ensures flexibility which in return ensures better control.
	Public by default. Blockchain could be private if used by companies (i.e., not for public need).	Always private.
	Decentralisation of nodes and miners.	Full control (see below).

	Blockchain	ID Ecosystem
Control	Wallet address doesn't store any actual data. It only keeps the links to the data stored elsewhere. Users' data doesn't sit with them.	Users give access to apps to use certain data in their ID as opposed to apps giving users access to the data they store for them. Users can manage access to apps at any time. If a user never uses an app again, the data will always be with him and he will not leave any data in any unwanted places.
	Non-programmer users can't create crypto assets and smart contracts by themselves.	Non-programmer users can create any assets or contracts in seconds by themselves without the need of any third party.
	You can't delete data on blockchains. It will always be there. If someone manages to get hold of your seed phrase, they can always access the data. If a person passes away, his/her data will forever be on the network with a seed phrase that can decrypt it.	You can delete the data in your ID. A person can even write a "will" that will trigger the deletion of the data with erasure at a given time in the future.
	Re-keying of the seed phrase is not possible. If the seed phrase is compromised, users need to move all the assets to another wallet as quickly as possible.	Re-keying is possible.
	You can't migrate your wallet address to another network. For example, you can't use your ETH address on Solana network.	You can import, export or migrate your ID and data as you like.
	Zero knowledge is optional and rarely utilised as the data is public.	Zero-knowledge is inherent in the ID ecosystem. Assets are encrypted with unique keys under the control of the user.
Decentralisation definition	Blockchain networks believe that active decentralisation is a **prerequisite** for data ownership and control. Decentralisation is mainly defined as the **decentralisation of the machines** that process that data.	IDZ believes that decentralisation is a **passive outcome** for ownership and control of data. Decentralisation is mainly defined as the ability to conduct **any transaction peer-to-peer**, without the involvement of any external parties. It implies centralising all the powers in the users' hands.

Crypto wallet vs ID:

	Crypto wallet	ID
Default name	Wallet addresses are impossible to remember, and users need to copy and paste them.	IDs are memorable by default. However, user can choose an address-style ID to make it impossible to remember.
Vanity names	Can be purchased as an NFT.	Can be purchased as an asset.
Identity concept	Avatars to be used in the Metaverse and unique usernames.	You are your data. ALL your data IS you. Not limited to just avatars and usernames. IDs are for the real world, not only the Metaverse.
Identification	Federated (using wallet address).	Federated (using the ID).
	A signed asset can be used as an identifier (e.g., using a certain asset to grant access to an account/channel).	'Identify' and share an Asset as an identifier (e.g., using a certain asset to grant access to an account/channel). IDZ envisages that the terms & conditions of websites and apps will become a contract which governs the exchange of certain assets between IDs (between the user and the service).
Offline accessibility	Not applicable.	You can encrypt your files and save them locally only. You can always access your local file even if you are offline.
Portability	Wallet addresses are tied to the network. For example, you can't migrate your Solana address to use it on the Ethereum network.	IDs are not tied to the ID provider. They can be portable given that ID clouds have the same structure.
Purpose of use	Wallet addresses are developed to be used with decentralised networks only.	IDs are developed to be used with all services, websites and apps, centralised or decentralised.
Corporate ID management	Not applicable.	Corporates can create IDs and assign them to employees. These IDs can contain all the necessary data for the employees to perform their duties (files, access to apps, access to offices, business expenses, etc.). Check the employment agreement and Optimal corporate experience: https://idz.com/use-cases/employment-agreement https://idz.com/use-cases/corporate-experience
Funds excessiveness & leftovers	To purchase tokens, you need to have excess funds in your wallet to proceed. So, if an asset is $10 and the gas is $0.5, you need $10.75 to proceed.	To purchase an asset, you need the exact amount of currency asset in your ID to proceed. If an asset is $10, you only need $10 to proceed.
	After purchasing an asset, users are often left with leftover funds in their wallets which can't be used.	You can use a definite number to buy a fixed price asset without leaving any leftovers in the ID.

Crypto assets vs assets:

	Crypto assets	Assets
Nature and type	Crypto assets are public and unstructured. They need programming knowledge to be created unless a platform (e.g., OpenSea) is used to create them. This limits the size of the asset universe.	Assets are private and structured. The structure is public for all (see below). No need for programming knowledge to create any kind of asset. Consequently, the asset universe is unlimited. Anything can be an asset, whether it is public or private. For example, user's music files, consultancy services, pictures, browsing history, health data from wearables, medical records, etc. Most notably, there is a chat asset so users can own and control their communication.
Hardware as an asset	Not applicable.	Assets can be anything even hardware devices. 01 watch is the first hardware asset in the ID ecosystem. Users need to own the digital asset for the physical watch to be active.
Asset connectivity	Assets are independent of each other.	Assets talk to each other. If you make changes to one asset, all the other assets which are linked to it will change too. For example, if your bank and school get your address from your utility provider (e.g., electricity company), they can use it in their assets. A change in the address held by the utility provider causes the address to change elsewhere. Refer to the hotel example: https://idz.com/use-cases/hotel
Structure	Assets don't follow a public structure.	Asset structure is public, allowing users to: • Create standard assets by simply populating fields (e.g., loyalty cards, tickets). • Request specific properties in known assets (e.g., requesting age property from the driving license asset).
Biometric assets	Not applicable.	Biometric assets ensure that only genuine owners can activate and use these assets. No one else other than the genuine user him/herself can use these assets. This has the following benefits: • Official identification: passport use case. https://idz.com/use-cases/official-documents • Binding agreement: digital signature use case https://idz.com/use-cases/secure-signing • Apostille service elimination: no need for official witnesses anymore.
Mutability	Not applicable.	Yes, if desired. For example, the fitness asset gets updated on a regular basis by the data it receives from the different wearables that collect the user's fitness data. Refer to the: "at the doctor" use case: https://idz.com/use-cases/doctor
Asset as a message	Not applicable.	Users can chat with each other by creating chat assets. In the chat asset, they can send and receive text, files, assets and contracts.

Crypto smart contracts vs contracts:

	Crypto smart contracts	Contracts
Structure	Crypto smart contracts rely on code. Programming knowledge is necessary for users to be able to build them. They are complicated for non-programmer users.	ID ecosystem contracts rely on pre-built formulas and models. Users can select from the dropdown lists the IDs and assets involved and build contracts in seconds.
	For two parties who don't have any programming experience, a third external party is a must.	No external third parties are required when building a contract between two parties who don't have any programming knowledge.
External information in the contracts	Users can use Oracles to insert external variables in their code which would affect the valuation or execution of the contracts.	Users can use the Key-value-pair section to insert variables in their formulas and models which would affect the valuation or execution of the contracts.
	Oracles are complicated for non-programmer users, as they need relevant coding experience.	Anyone can use and developers can contribute to the key-value-pair page
Transparency	Crypto smart contracts are complex for non-programmers. Only developers can know the possible outcomes of a contract if they decide to dig into the code.	Contracts are transparent for all users. No grey areas in the middle. Users can know all the possible outcomes of a contract in advance.
	Given that it is not necessary for the contract code to be published on blockchain explorers, users must trust the developer who created the contract.	For example, if a tenant wants to check all his rights before and after signing the rent contract, he can simply check all the formulas included in the contract, what he can and can't do, and what the landlord can and can't do, all without needing to read a whole legal agreement.
Mutability	Crypto smart contracts are immutable. For example, if a contract between an employer and employee needs to be amended to reflect a pay rise, a new contract needs to be written from scratch.	Contracts are mutable with the consent of all parties involved. For example, if a contract between an employer and employee needs to be amended to reflect a pay rise, they can simply amend the old contract. Users can also build immutable contracts if they wish.
	Ethereum supports smart contract versioning.	ID cloud providers support contract versioning.
Escrow and binary execution	Non-programmer users can't check which assets the contract requests and/or sends to them. Many users fall victim to scam contracts which drain all their wallets without sending any assets in return.	Contracts act in a binary manner that ensures justice for all parties involved. For example, check the exchange use case: https://idz.com/use-cases/exchange
Contract as a message	Not applicable.	Users can chat with each other by creating chat assets. In the chat asset, they can send and receive text, files, assets and contracts. Contracts can be executed on the spot inside the chat.
Privacy	Crypto smart contracts are public. For example, if a user sells his house, all users could know about it.	Contracts are private. For example, if a user sells his house, the information is kept private, not available to the public.

	Crypto smart contracts	Contracts
Speed of execution	Crypto smart contracts are slow as they need to be verified by different nodes.	Execution is instant.
	Users can get scammed by a contract that requires all assets at once, as they can't verify the code and the assets involved.	Full execution of the whole contract (regardless of the number of assets involved) is one click away. Users can check all the assets involved before they share. Check the cinema use case.
Identity verification in contracts	Not applicable.	Biometric assets can be used to sign contracts. This ensures users it is really the genuine people who are signing the contract as they are the only ones who are able to use these assets. For example, a user can use a biometric passport asset issued by a government body which only s/he can activate/use to sign a contract.
Contract execution by hardware	No	Yes. For example, a user can unlock his/her car (i.e., execute the contract that governs locking/unlocking the car) with his 01 watch.
Payment reference & discount coupons	There is no payment reference, so users rely on the transaction hash to confirm the payment. It will not be viable to rely on transaction hash if blockchain were to be adopted by all people. Some services offer "notes" with payment, these notes are usually public and subject to abuse.	Payment references are available and private
	Users can't apply discount coupons for the assets they want to purchase. Early adopters of a token can't benefit from a discount that applies to them only.	Discount coupons are available and private.