DUND

A public permissioned WRKChain ecosystem built for Enterprises

To give context to the problem being solved, one should understand that DUND began as a “2nd layer” protocol – drawing from the core team’s collective experience in enterprise consulting and software development, DUNDwas originally designed to be solely a “data liquidity” protocol which would allow enterprises to standardize/tokenize their data and place it in a liquid format to be bought/sold/transferred over a blockchain. What we were building was not “sexy,” rather it was fulfilling a very lucrative, yet possibly boring, need for enterprises and the future of data.

The team spent the better part of a year developing the codebase and engaging in early enterprise outreach – over the course of hundreds of conversations and then subsequent verification of realities, it became quite apparent that it was not possible to create a functional “2nd layer” when in practical reality there was no “1st layer” that worked in any way that would be functional for any sort of serious enterprise trying to do “work.”

By “work” we mean the daily mundane – sometimes automated execution of smart contracts that happen in the thousands/tens of thousands/millions and are not directly related to any sort of immediate monetary gain/loss of money/tokens. Early concept chains such as ETH and EOS gained prominence in 2017-18 and laid out a vision of how things “should be,” but limitations were quickly exposed when it became apparent that current technological constraints would not allow “all the smart contracts in the world” to be validated by any single blockchain.

Even with the best of intentions, when placed in an “n+1” scaling situation, any “closed” system quickly achieved critical mass, and all of the “work” transactions got pushed to the back while “high value” transactions such as “DeFi”/coin speculation (ERC-20), gambling (EOS), and pseudo-gambling (CryptoKitties, etc.) would by nature clog the network.

The other side of the solution that has been approached from the corporate side has been the deployment of fully private “consortium” blockchains, as proposed by offerings such as Corda and Hyperledger. Being built on the “linux/redhat” model, the idea was to create open-source software that “does useful things” and then bill for consulting to implementing and maintaining. This would effectively and instantly solve the “scalability” issue as the required amount of validators is (n) – i.e. whatever the deployer determines, and only transactions allowed by the validators will be allowed on the network.

This means that these fully “private” implementations are useful in theory but, depending on the use case, can sometimes be regulated to nothing more than a glorified database so that someone can say, “We are on the blockchain.”

The Solution

Evolving to the “useful” endgame – DUND has been through numerous iterations, all of which have been built on each other and, through each phase, we continued to ask ourselves the obvious but often muted question – “How does this benefit our user?”

To answer this, we engaged in hundreds of conversations with existing enterprises/governments about how the blockchain can practically benefit their endeavors. Three practical examples and requests that came across our desk were:

- A government bank in a developing country wants to build a stablecoin – this stablecoin would be issued to all agencies to deploy for contracts awarded. Contractors would be paid in the stablecoin and would need to redeem it at the government bank for fiat. The purpose of this is to monitor government payments and cash flow in an ecosystem where corruption and fraud are abundant. For this stablecoin, what is important to the government is fast, consistent, and extremely inexpensive transactions. The actual full public consensus from a system such as POW is not as important as the trust entities within their network to run the validation nodes – however, users and agencies will need full public transparency in order to trace the history of transactions.

- A gaming company is building a card game where each card will be non-fungible and tokenized, and in-game currency will allow it to be traded for cards and/or loot boxes. Most users won’t even understand they are on the “blockchain” and can’t be expected to hold/stake a wallet balance in order to execute smart contract transactions. However, they want these cards and tokens to be publicly tradable and accessible. Transactions need to be fast, consistent, and free for the end-user.

- An umbrella organization of clinics in a developing country with over two million yearly patients on record wants to place all of their clients’ data (EHR) on the blockchain to be accessible via a data wallet. The end consumer will be able to access and deploy their data to different providers but can’t be expected to hold and operate a “speculation coin and private key” in order to access this data. This group of clinics will be the beginning of a consortium organization and handle the validation themselves. Later, when other clinics or entities join the consortium, they will bring over their user-bases along with joining the consortium to participate as a validator.

These examples are just three of dozens that we have laid out as business models and implementations – speaking with all of the governments and enterprise clients, we have learned that far beyond a simple majority all have the same basic needs/requests for a blockchain for their specific usage:

1. To be fast and consistent – they should not have to compete with entities outside their consortium for network resources.
2. Transactions essentially need to be free or nearly free (pennies of pennies) – if there is a transaction cost, then users of the blockchain should not be expected to have to pay or stake to execute these transactions.
3. None had a particular desire, need or allegiance to use a “global” coin or token for daily transactions (such as ETH, BTC, or DUND) as they can have fluctuating prices. Most requested to use their own internal coin to be used as a marker for transactions. This internal coin (if they even choose to issue one) could be stable or fluctuate depending on the parameters set by the validators.
4. On the same note, most wanted a mechanism of interoperability where there would be the ability to change fungible/non-fungible assets into other implementations on the global DUNDecosystem under the correct circumstances.
5. All had no problem handling their own validation methods and had no desire to subject their projects to outside validation and costs.
6. On the same note, all supported an outside oversight mechanism/auditor which would stamp “trust” on top of their validation so that users could gain trust.

There were also various opinions on network/data accessibility and privacy. Some would want the data transacted to be fully, publically traceable and transparent – others would want it encrypted to protect privacy or locked behind firewalls like a private database. In these requests, all implementations are possible.

Working with our clients on a daily basis, and seeing their needs and the reality of business, led us to an understanding that there is an answer to the quintessential question of scalability of blockchain that involves a hybrid public/private approach which encompasses the DUND ecosystem.

Introducing DUND

There are two aspects of DUND – we can call them simply “Mainchain” and “WRKChain,” which can be more commonly understood as in the same category as a “sidechain.” Let us break down their responsibilities.

At its base, the Mainchain is more or less what you imagine what a blockchain to be. It is a BFT-based PoS chain with a native currency called DUND.

Governance of the Mainchain is executed via DSG (Distributed Stake Governance), which will be explained in detail further. However, the summary version is that there are 96 EVs (elected validators) who maintain the network and collect tax for validating blocks. To become an EV requires a vote and stake. This effectively means that to control a lifetime node, one would have to accumulate approx 1.04% of the token supply and actively place that on stake – this is assuming all tokens are circulating and fully staked for voting.

DSG has been designed so that the number of resources required to effectively, hostilely take over and disrupt the network would by nature put the attacker in a position where they would be inherently vested in the success of the network.

Simple enough?

Now if we had a world where there was only the Mainchain – we would effectively have "Tendermint with DSG governance instead of POW" – which may be considered an incremental improvement with faster TPS due to the reduced amount of validators (96 at a time as opposed to 10k+ with PoW). However, many projects are attempting something similar and ultimately without a major technological jump, meaning previous adoptions and comforts will win the day.

Where it starts to get fascinating is when we go back to these conversations where we understood what enterprise entities truly need in a blockchain. These are the entities that need blockchain to do "actual work" that is not directly related to direct monetary gain per transaction and need that blockchain to produce this work in a consistent and price-controlled manner.

As we discussed earlier, private implementations such as Hyperledger Fabric can theoretically solve this – however, they are implemented in closed gardens with no oversight on the transparency, and no possibility of interoperability.

The solution is a variation of "sidechains" which we are calling "WRKChains" that take a public/private approach to execution.

When an entity wishes to open up a WRKChain, they upload to the Mainchain – along with a payment of DUND – an "expansion log" following a standard. This is more or less technically equivalent to a smart contract – this expansion log will state the number of validators in the WRKChain, how these validators are selected, details on the token being generated on the WRKChain (if they decide to have one) and how tax is paid to the WRKChain validators (if it is even taxed via the native WRKChain token).

The expansion log also describes how the WRKChain will "check-in" with the Mainchain. Typical usage will find that WRKChain validators will work in their own trusted vacuum – validating transactions with the WRKChain coin and broadcasting the headers of each of their blocks to the Mainchain, which then includes this information in its own trusted block.

The nature of this system assumes that we cannot guarantee that the information produced by WRKChain validators is immutable, as they are operating typically on a dPoS/Federated basis. However, the Mainchain can verify each header they submit and guarantee that they have not changed information or reorganized previous blocks due to a Merkle tree root.

WRKChain validators *are able to:*

- Process intra-WRKChain smart contracts and token transfers of the WRKChain tokens.

WRKChain validators are *NOT able to:*

- Control or submit changes for transfers of DUND, as DUNDis controlled only by the fully trustless DSG consensus of the Mainchain.

This creates a practical ecosystem where the Mainchain is the “Blockchain of Blockchains” – allowing independent entities to deploy useful scaled solutions that share the trust and interoperability of the Mainchain while maintaining the speed and scalability of private implementation.

This is a good theoretical start to solving a complex and universal problem – however, the reality of usage and uptake is in the technical details, which we will now explore.

Distributed Stake Governance

1. EVs are responsible for validating the DUNDMainchain and producing blocks according to a consensus model. Blocks are currently set to produce every ~5 seconds and EVs are rotated with respect to block production until a consensus is finally reached.

2. The EV, upon producing a successful block, will receive the most block rewards according to the tax submitted. All of this DUNDis placed in a wallet that the Validator and any of its delegators can withdraw.

3. To be an EV is not a technically intense operation, and EVs are not expected to execute any other role other than to validate the network and collect tax.

The purpose of this system is to create an ecosystem where the staking of DUNDtokens for the top 96 is rewarded with interest from the job of being an EV, while immutably securing the network.

Circulating Supply and Network Tax

1. It is important to note that during our testnet phase (2018-19), the foundation had minted 1,000,000,000 “UND” testnet tokens in a dual ERC20/BEP2 format. Upon mainnet, these “UND” were retired and removed from circulation.

2. In May of 2020, the foundation released its mainnet and the Mainchain DUNDtoken, of which 120 million were minted.

These are the following tokenomics on day 1 of the mainnet:

- 120,000,000 total supply
- 120,000,000 circulating supply

While there is no inflation to stimulate node operators, they are rewarded by network tax collected from WRKChain submissions. This is very important because it forces a useful ecosystem where nodes are only paid if they are providing value.

With regards to an increase in supply, the foundation may issue up to 10 million locked DUNDs per year for up to 9 years – bringing the maximum lifetime supply to 210 Million.

These tokens are the same as any other DUNDtoken except that they will be locked, such that they can only be used to pay network tax for running beacons or WRKChains. This means that these tokens cannot be resold, staked or transferred.

Once a validator processes a transaction with these locked tokens as tax, the validator will then have ownership of the tokens in their wallet and will be able to transfer them freely.

This dynamic not only ensures a predictable price schedule for the users of the network but also provides an ongoing revenue stream for the foundation, as history has shown that “one time raises” typically lead to failed projects, and that anything good can generate ongoing interest and DUNDing for the value it provides to the community.

The Mainchain

The Mainchain is the backbone of the network is serviced by EVs, which are selected according to the DSG process outlined above. The Mainchain’s primary role is to increase the trust of a WRKChain by recording a submission of its headers, and maintaining a chain of those recordings – which is accomplished by acceptance of data deposits from any WRKChain that it currently services – and additionally facilitating the native transfer of DUNDbetween accounts on the Mainchain.

The Mainchain will not have native support for dApps/generic smart contracts, which will require their own WRKChain in order to run. The reason for this is that the DUNDMainchain is not designed to be a “global supercomputer” – rather, it is a trusted entity that validates and collects tax from entities in the ecosystem.

Each WRKChain will be rooted in the Mainchain with its own expansion log that will contain the WRKChain’s data deposits in addition to WRKChain metadata, which is discussed in more detail below.

Expansion Logs

Expansion logs are the method by which the Mainchain can store and track the state changes made by a WRKChain. When a WRKChain produces a new block, it generates a transaction with the Merkle roots for the current state of the WRKChain, the block number, header hash, and any signatures of the WRKChain block validators (if applicable).

WRKChain’s can submit the block hashes to the Mainchain, where it is stored within the Mainchain’s WRKChain root module.

Proof of Genesis

When a WRKChain is initially deployed, it may store a hash of its genesis block in its Mainchain root. This further enhances the immutability of the WRKChain and allows the full history of the WRKChain to be audited via the Mainchain, right back to the point of genesis.

WRKChain ID

Upon registration, each WRKChain will be assigned a unique ID, the master record of which will be stored within the WRKChain’s root on the Mainchain, and perhaps also an easy to read alias

WRKChains

WRKChains can be deployed by anyone. WRKChains deployed using the SDK provided by DUND Foundation are fully EVM compatible chains and can therefore support any dApp or smart contract, and may have their own native coin (or not). SDK eases the deployment of a WRKChain, however, there is no restriction on what it can be and is not limited to EVM-based chains. WRKChain operators may define and implement their own consensus methods and

transaction fee structure, or adopt any popular chain such as Tendermint, Corda, Hyperledger, etc.

Each WRKChain will be rooted in the Mainchain with its own expansion log. When a WRKChain adds and finalises a new block to its chain, it will also be responsible for generating the required transaction to call its root, anchored on the Mainchain, to which it will send its expansion log data deposit (Merkle roots, block headers, etc.).

Depending on the level of validation WRKChain operators require, they will be able to define whether they require every new block validation or less frequent intervals (for example, every 10 or 250 blocks require validation by the Mainchain). The higher the level of validation required, the more the Mainchain network taxes the WRKChain operators, who will be required to pay.

Beacons

Beacons are to be considered a value-add Timestamp SDK, which can be applied to most modern databases. A beacon can work in two different ways, depending on the needs of the user.

OPTION 1:

A - Every change to the database is recorded and that change is encrypted into a hash.
B - Every XX minutes, these hashes are used to generate a Merkle tree and sent directly to be recorded/time stamped on the Mainchain, along with a small payment of DUND.

OPTION 2:

A - Every XX minutes, a snapshot of the DB is recorded and condensed into a hash. This is then sent to the Mainchain, along with a small payment of DUND.

Note: Due to the potential for the extreme size of databases being hashed, the user will be able to define which critical data they want to timestamp.

The purpose of a beacon is to add a layer of immutable trust to existing or newly built centralized database projects.

By timestamping with a beacon, an entity can be guaranteed that information has not been modified by an unseen force, hence enhancing security and immutability. DUND will make available an API that will allow nearly any sort of entity or configuration to hash and timestamp whatever information they please.

Project Ethos as a Pure Utility

The DUND token is designed to be used as a utility for tax on the Mainchain,
with ongoing DUNDing for the project through selling tokens into locked wallets.

The purpose of DUND is to create a public/private hybrid ecosystem that allows operators of WRKChains to control the cost and speed for themselves and their clients while sharing the immutable trust of the Mainchain. We believe this is the solution blockchain has been waiting for.