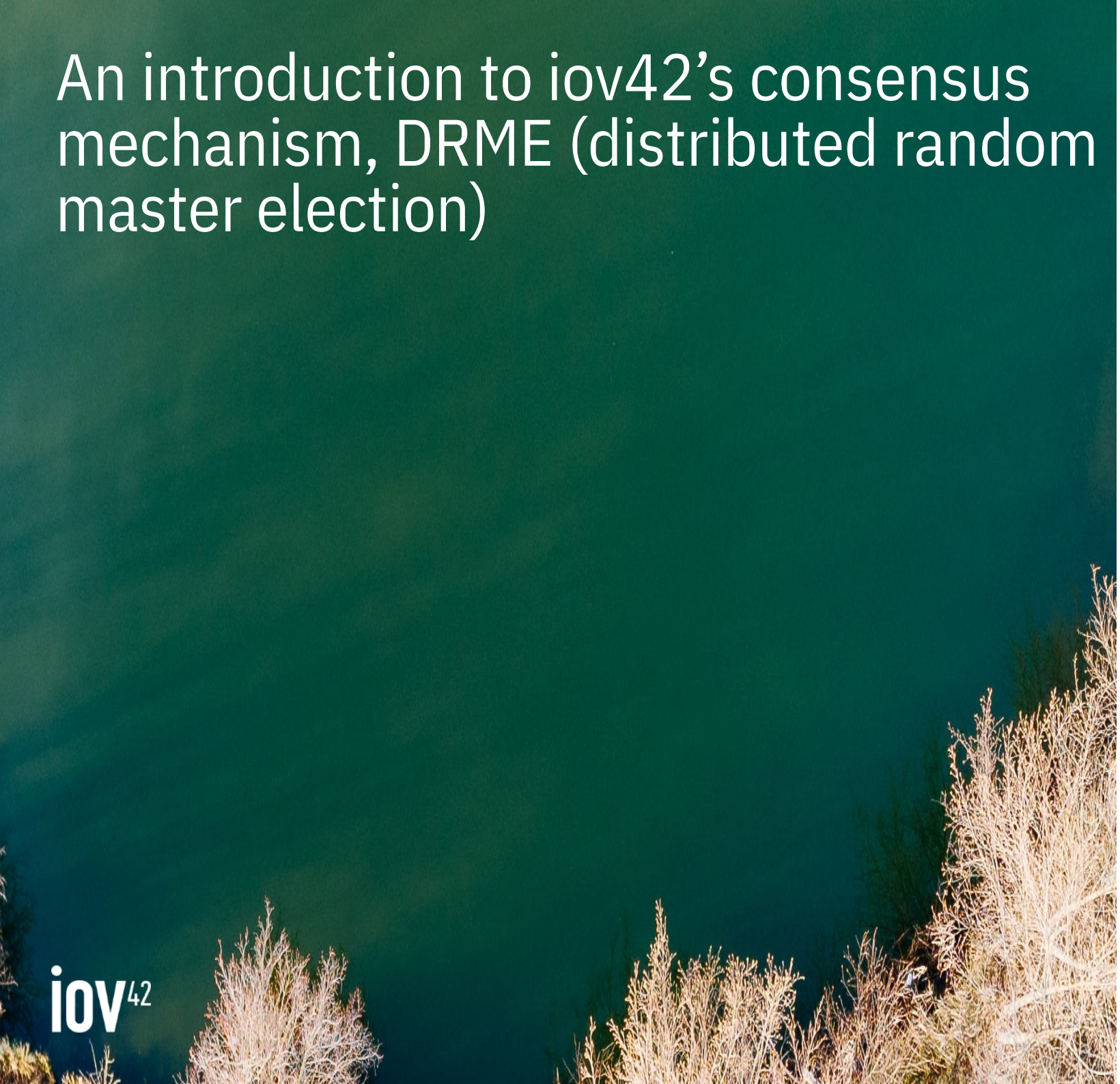
An introduction to iov42's consensus mechanism, DRME (distributed random master election)



One of the most critical aspects of any distributed ledger technology (DLT) is its consensus algorithm. Consensus algorithms enable transactions to be authenticated and validated on a distributed ledger without the need to trust or rely on a central authority. It is by consensus that the parties of a consensus network agree on what the "'true" state of the system is at any point in time. When consensus is reached on the validity of a transaction, it will be committed (added) to all copies of a distributed ledger. A consensus algorithm can dictate the security, performance, and environmental impact of a DLT.

Over the past few years, new and evolving consensus algorithms, as well as combinations of algorithms, have made the next generation of secure, scalable, and interoperable enterprise-ready DLT solutions possible.

DRME: a novel approach to achieving Consensus

In an iov42 network, consensus is responsible for protecting anything that requires trust—from users making claims about their identities to exchanging the ownership of an asset on the network. Ultimately, no changes can be made to the state of the system without going through consensus.

iov42 has developed its own consensus mechanism called DRME (Distributed Random Master Election). This proprietary consensus algorithm leverages the architecture of the network to be highly-performant and energy efficient, while being able to randomly assign which nodes lead the coordination of any given consensus decision on a per transaction basis.

The DRME process is divided into three key parts:

1. Master election
2. Voting to reach consensus
3. Committing the transaction to the network

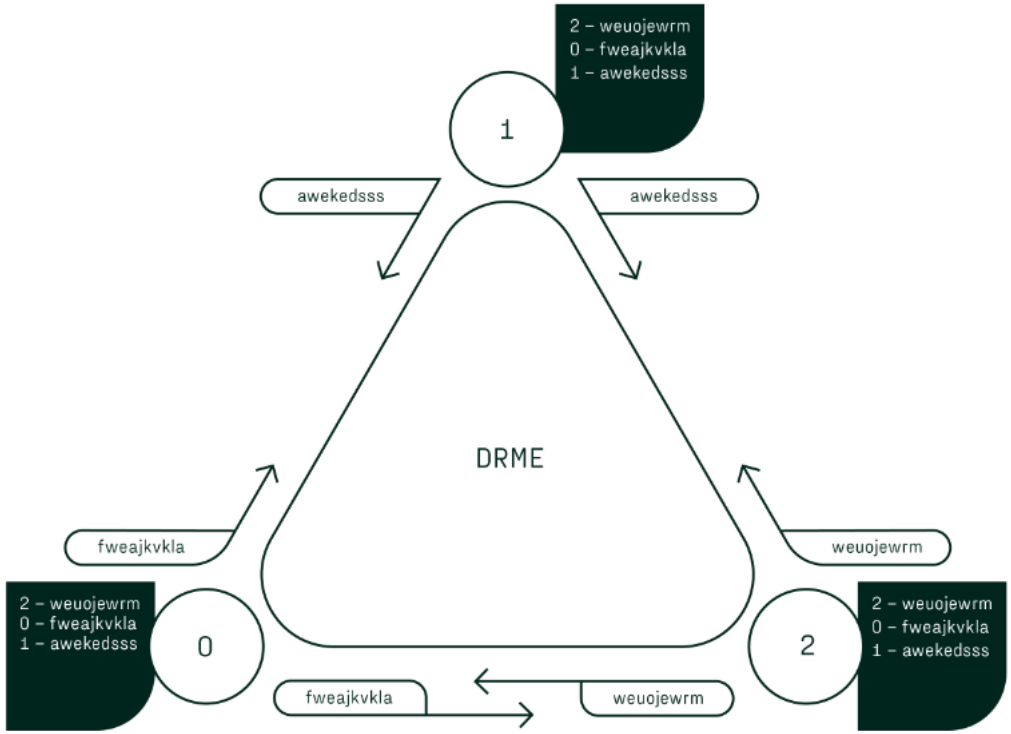
For each transaction that passes through the platform, one of the consensus participants is randomly elected to be the “master” that will coordinate the voting process amongst all network participants to determine whether the transaction is valid or not. Based on participant votes, the master ultimately decides if the transaction is valid and can be committed to the network’s immutable ledger, or if it is invalid and must be rejected.

Master Election: DRME

iov42’s consensus process starts with its master election algorithm, DRME, which is where the entire process gets its name from. DRME is the process that establishes which participant decides the validity of each transaction. The goal of this process is to be completely random, as well as agreed upon by all participants.

DRME starts with a series of messages known as DRME fragments. Each fragment contains a completely random chunk of data of a known length. All participants within the network generate and broadcast these random fragments to all other participants at regular time intervals. The received order of these messages is identical across the network, so each participant has the same understanding of the state of the system.

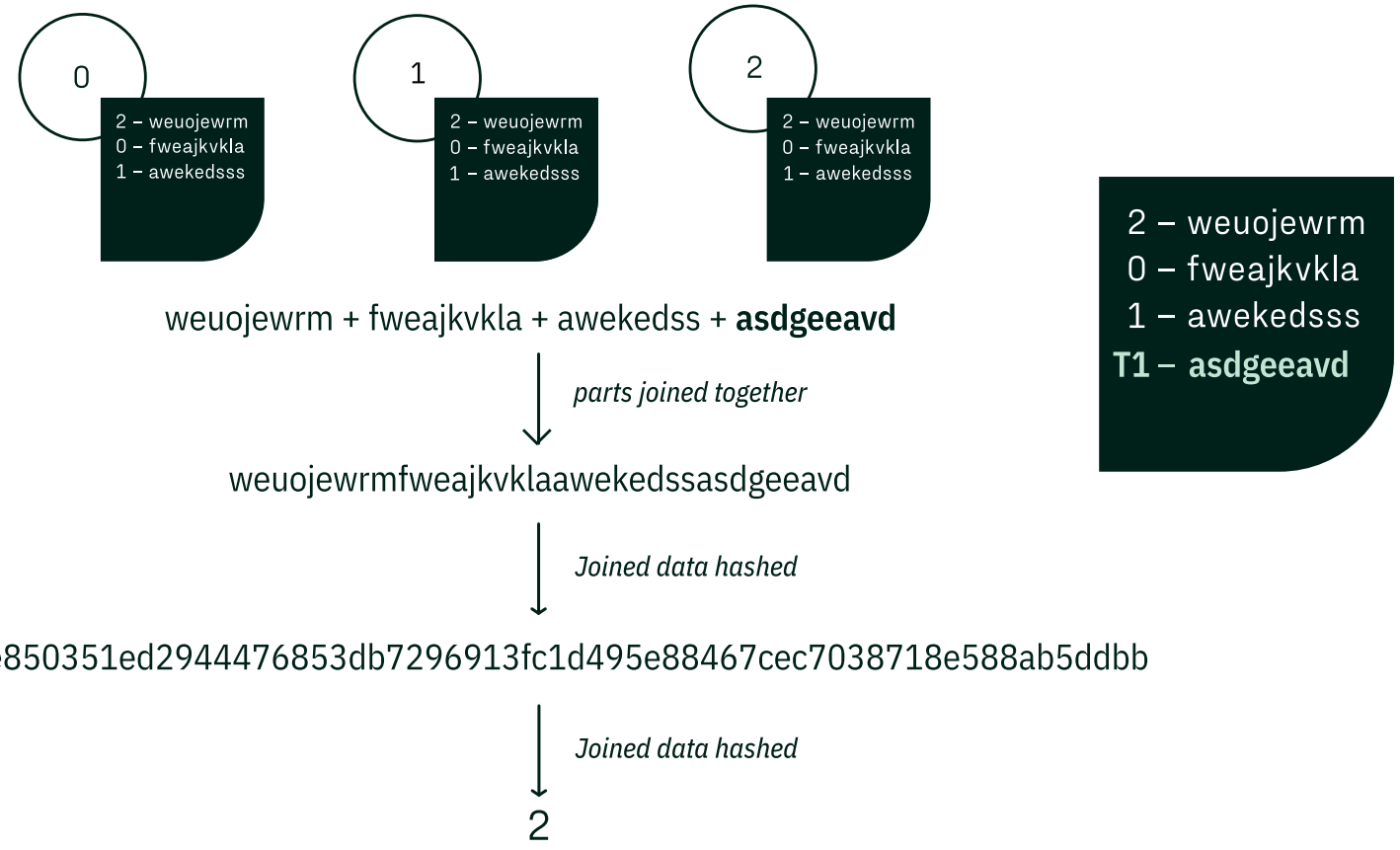
In the example to the right, there are three participants in the network. All of the participants are sending DRME fragments to each other at regular intervals. The order in which the fragments are received is random, but deterministic, and reflected in a message log. In this example the message log shows that the fragment from Participant 2 was received first across the network followed by the fragments from Participant 0 and Participant 1. The participants will continue to transmit DRME fragments at regular intervals.



Incoming transactions are broadcast to the network participants through the same path that all DRME fragments are broadcast. This means that fragments and incoming transactions are interspersed, ensuring that each participant has the same view of the world when a transaction needs to be validated.

When a transaction arrives, the latest fragment from each node is joined together with a unique value from the transaction. This combined fragment is first hashed using SHA-256. The resulting hash value is then passed through a modulus function, based on the number of participants. The result of the modulus function determines which participant will act as the master for that transaction.

In this example to the right, a new transaction has been received. The latest DRME fragment from each participant is ordered and combined. Then the unique data from the transaction —in this case, "asdgeevd" — is added to the string of combined fragments. All of these combined parts are passed through a hash operation to produce a value that is then passed through a modulus function. As mentioned above, the modulus function is based on the number of consensus participants, so this example uses



MOD3, since there are three participants. In this example, the modulus function produced a value of 2 which means that participant 2 will be the master for this single transaction.

Reaching Consensus

The role of the master is to coordinate the activity for any given transaction. The elected master asks all participants, master included, to vote on the validity of the given transaction. Each participant will check that the request is valid based on its understanding of the state of the system (e.g. “is there enough balance to perform the transaction?”).

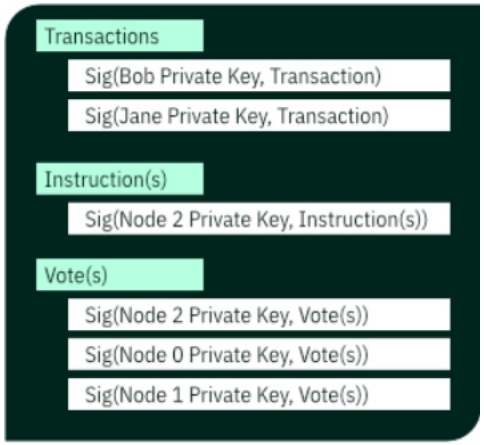
Throughout this consensus-reaching process, integrity is ensured by the use of digital signatures at each step. Each transaction that is submitted must be signed by the identity that submitted it. The master signs its instructions before sending them out to the participants to prove it was the source of these instructions. Each voting participant signs its vote to prove it has voted. All signed votes are collated by the master.

Committing the Transaction

Once the master collects the votes from all of the participants, it issues the appropriate action based on those votes. Currently, the decision is made based on a majority vote. However, in the future, some transactions may be able to require a unanimous vote.

If the system participants concur that the transaction is valid, the master will issue a commit message. This message takes the form of a transaction proof that contains all of the steps and signatures associated with the transaction. The proof has been rendered to the right for demonstrative purposes.

This proof is then signed by the master to prove it is the collator of the proof. Finally, the proof is committed to the network's immutable distributed ledger.



Why is all of this Important?

While the randomness of DRME ensures that a bad actor cannot manipulate a iov42 network, the protocol's per-transaction approach allows the global iov42 platform to combine high performance, security, and low energy consumption, positioning iov42 as an ideal DLT solution for enterprises and governments.