Transactions from the PoV of tendermint & LL-core

Tendermint nodes only communicate opaque blobs (Tx) and ask the app via CheckTx if the blob goes into the mempool (for inclusion in the block or not).

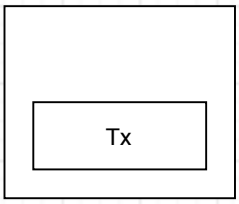
Note that execution of the included Tx (and modification of the state) happens "one off" after the Tx were included in the block; via DeliverTx

Although we want to get rid of this, we will unlikely launch our first dev-net with "immediate execution"

This is the current format of Tx.

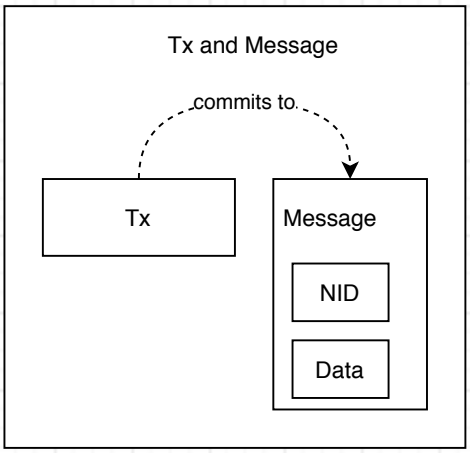
Currently

- Tx are just opaque blobs from the PoV of tendermint.
- It doesn't even (need to) know the internal encoding of those
- the structure and meaning of the Tx is defined somewhere else (in the App)



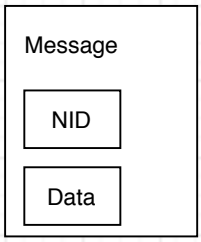
Pair of Tx & Messages

- if tendermint sees this it can detangle Tx and messages



Messages all the way down

- Here there are only messages
- and Tx are just a special kind of message
- all messages need a namespace field (including state relevant Tx)
- messages and tx are treated basically the same way
- their namespace IDs can be used internally
- this also means changing above protobuf message to contain a namespace (as opposed to just the raw tx data)
- Tx and Message pairs are communicated separately here



Pair of Tx & List of shares

- tendermint sees Tx and messages are already split up into shares
- to get the message back it needs to merge the shares