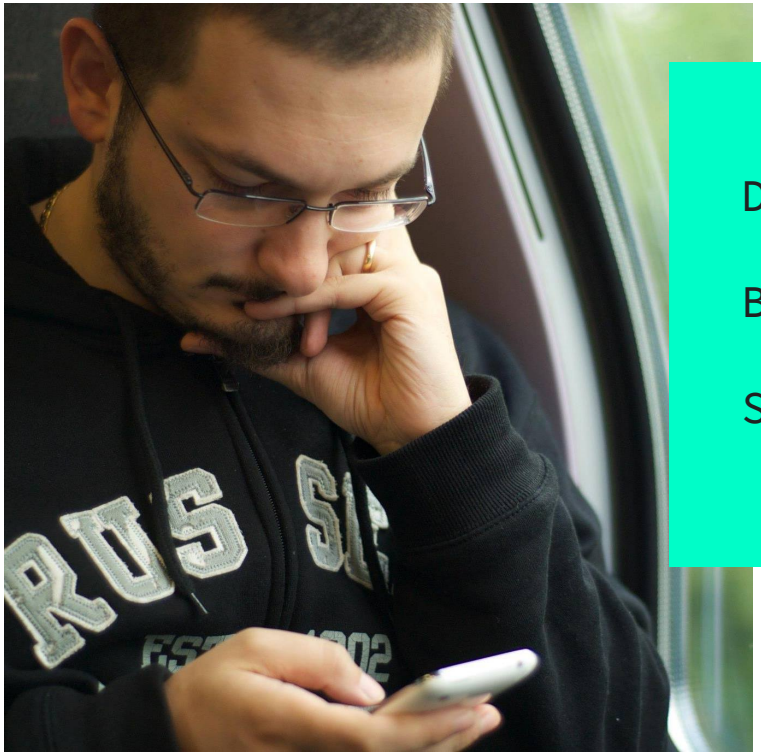
EIP-2848

My Own Messages

Who I am



Developer and tech enthusiast

Building in the space since 2015

Still learning

Peek me on the web

GitHub	https://github.com/neurone
LinkedIn	https://www.linkedin.com/in/giuseppebertone/
StackExchange	https://stackexchange.com/users/1844613/giuseppe-bertone

Pain points

sharing ideas, opinions, messages are all still very dependent on centralized services (unreliable, non-censorship resistant, lack of single source of truth)

hard to understand if messages come from a specific author (developer, journalist, etc.) so it’s easier to scam people

The (trustless) world I want

as a developer

I can easily talk to my user base

I have control of my reputation

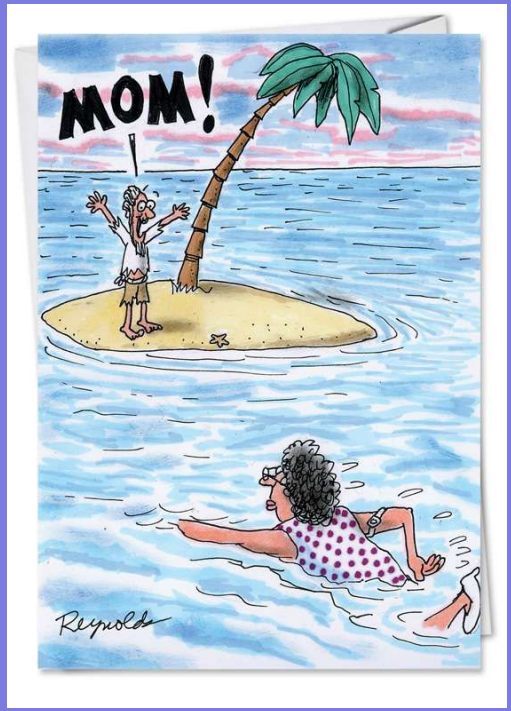
as a user

I can easily verify authors of posts, messages, articles

I can easily self-share my thoughts

I can control my content

Who ya gonna call?

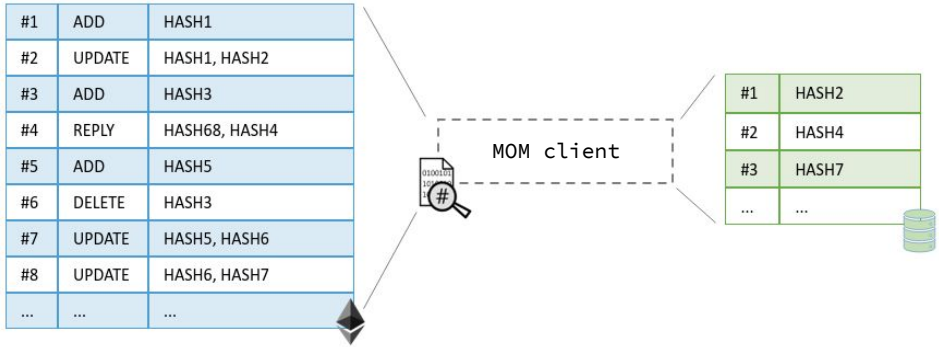


EIP-2848 is a standard to create your very own public, always updated, unstoppable, verifiable, message board.

How MOM works

Ordered list of messages

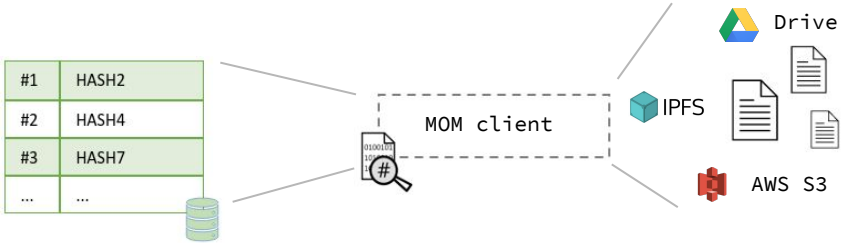
1. User creates a **message** and **corresponding hash**
2. User creates a **MOM payload** (op code + params)
3. User sends a **self-send tx with a MOM payload**
4. Client **reads all operations for an address**
5. Client **creates a local index of updated message**



MOM txs are very cheap: i.e. ADD operation with sha256 multihash -> 21,000 GAS by base Ethereum tx + 548 GAS by MOM tx = 21,548 GAS

Message data

1. User selects content to read, modify, etc...
2. Client **downloads the content from the persistent service preferred by the user** (i.e. IPFS, Swarm, AWS S3) CANs are preferred, but client knows the hash of the content so it can receive data from any sources



MOM Specifications

MOM TRANSACTION DATA STRUCTURE	
ATTRIBUTE	VALUE
TO	MUST be the **same account signing the transaction**
VALUE	MUST be **0 wei**
DATA	MUST be at least 2 bytes. The **first byte MUST be operational code** and following bytes MUST be based on the operational codes

MOM v1 CONTENT
A **multihash** represents **Markdown text in UTF-8 without BOM**, so clients MUST support content with **text/markdown** (RFC 7763) as media type.

CORE COMMANDS		
CODE	OPERATION	PARAMETERS
0x00	ADD	multihash
0x01	UPDATE	multihash, multihash
0x02	REPLY	multihash, multihash
0x03	DELETE	multihash
0xFD	CLOSE ACCOUNT	multihash
0xFF	RAW	raw

EXTENDED COMMANDS		
CODE	OPERATION	PARAMETERS
0x04	ADD & REFER	multihash, address
0x05	UPDATE & REFER	multihash, multihash, address
0x06	ENDORSE	multihash
0x07	REMOVE ENDORSEMENT	multihash
0x08	DISAPPROVE	multihash
0x09	REMOVE DISAPPROVAL	multihash
0x0A	ENDORSE & REPLY	multihash, multihash
0x0B	DISAPPROVE & REPLY	multihash, multihash

Payload sample: 0x0012200893a4130a758a891662112dbe16ddd0d6d131e15b379a50e7acec08ad941a36

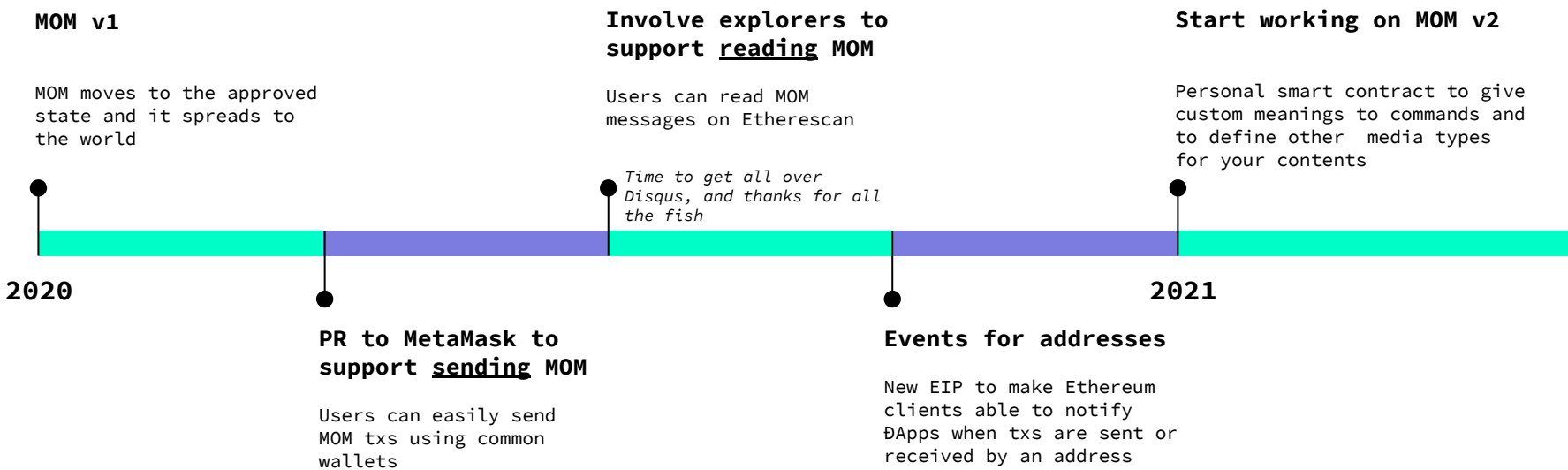
Detailed specifications in the official EIP repository: https://github.com/ethereum/EIPs/blob/master/EIPS/eip-2848.md

DEMO

mom-client v0.5.5

mom-js v1.0.4

What's Next


MOM v1
MOM moves to the approved state and it spreads to the world
Involve explorers to support reading MOM
Users can read MOM messages on Etherescan
Time to get all over Disqus, and thanks for all the fish
Start working on MOM v2
Personal smart contract to give custom meanings to commands and to define other media types for your contents
2020
PR to MetaMask to support sending MOM
Users can easily send MOM txs using common wallets
Events for addresses
New EIP to make Ethereum clients able to notify ĐApps when txs are sent or received by an address
2021

AMA

It means "love!" in italian

WHY IS EIP-2848 MOVING TO THE NEXT STAGE NOW?

- ★ **Easy to implement** in any language
- ★ **Immediate benefit** for all the community
- ★ **Tech is already available now**, no new technology to invent
- ★ DeFi is rushing and it urges an **easy** and **efficient** method to verify that communications come from developers and stakeholders
- ★ And, you can create a **lot of memes** about MOM

Why aren't you using a smart contract?

- ★ MOM txs are **much cheaper without using Smart Contracts**
- ★ Message **states are managed by the client**, you don't need a SC
- ★ You need to notarize content, you **don't need events**
- ★ You **cannot make mistakes** and MOM **works already on any network**
- ★ MOM v2 will take in consideration using a SC **to let authors define their own commands and rules**, but MOM TXs will continue to be self-send transactions

Why don't you support batch messages?

Batch and other techniques require a **pre-defined structure for the content**, but in that case:

1. Client-side parsing can introduce **endless loops**
2. Client would be **forced to download the all the content** to know the final state of the message board

For these reasons, MOM is designed to be able to determine the updated state of the message board by **just looking at the Blockchain**, and so you cannot have batch messages as a standard MOM message.

stay safe

Thanks for your time

only use nodes under your control