Goal

The goal is to prove the feasibility of a completely browser based P2P publishing platform that is distributed and not centrally moderated.

Hackathon Goals (Ryan)

the top three things i would like to accomplish this weekend are …
1) establish some mockups and maybe a basic working UI and what the product should look and feel like,
2) establish either what technology we should use in complete detail, or at least have some kind of technical direction (i.e., are we using ipfs, web torrent, starting from scratch, etc.?),
3) what is an example business plan for people such as myself who intend to run a node as a business

On GitHub -> https://github.com/dattnetwork/datt/
To be added to -> https://github.com/dattnetwork/dattdocs/blob/master/README.md

Principles

The following principles are essential to DATT:

- The network should be uncensorable and users must have final say in content filtering (or the lack of it)
- Strong anonymity by default, users may optionally assign a stickier identity to themselves via cryptographic signatures
- We aim for a truly decentralized solution without the need for central servers to hold any content or any registration of users; nor should they be aware of the specific content users are sharing and accessing.
- Nothing more than a web browser should be required to participate on the network
- A user should have plausible deniability that they have accepted/stored/consumed/propagated any content item

Secondary principles that arise from the core principles but are not essential:

- Content encryption should be pervasive until a user chooses to decrypt the content
- Publishers should be incentivised to create quality/popular content

MVP *(feel free to add/remove/move these points)*

It must be possible to:

- Post markdown formatted body text with a title | Post a link with a title
- Distribute post content between peers

- Sort content for display (chronological, total views, active peers, tips:peer ratio, average comment karma)
- Comment on posts and have the comments also distribute to peers
- Tag or Categorize posts

It is desirable to be able to:

- Monetarily incentivize contributors (Content creators)
- Optionally include a Title image
- Display only comments above a defined karma threshold
- Allow nested comment threads
- Throttle content submissions to reduce spam

It is not necessary to:

- Allow editing of the body text or title after publishing

Potential weaknesses

- If the cost of publishing is too low spammers may create a large quantity of posts even though they are unlikely to be seen. Client side hashing may not be the answer.
- All javascript will need to be open source and vetted for vulnerabilities.
- It may be difficult to convince a critical mass of users that this is sufficiently better than alternatives such as Voat and Aether.
- If there are no pseudonyms so the community may have less social fabric than other platforms.

Thoughts and Ideas

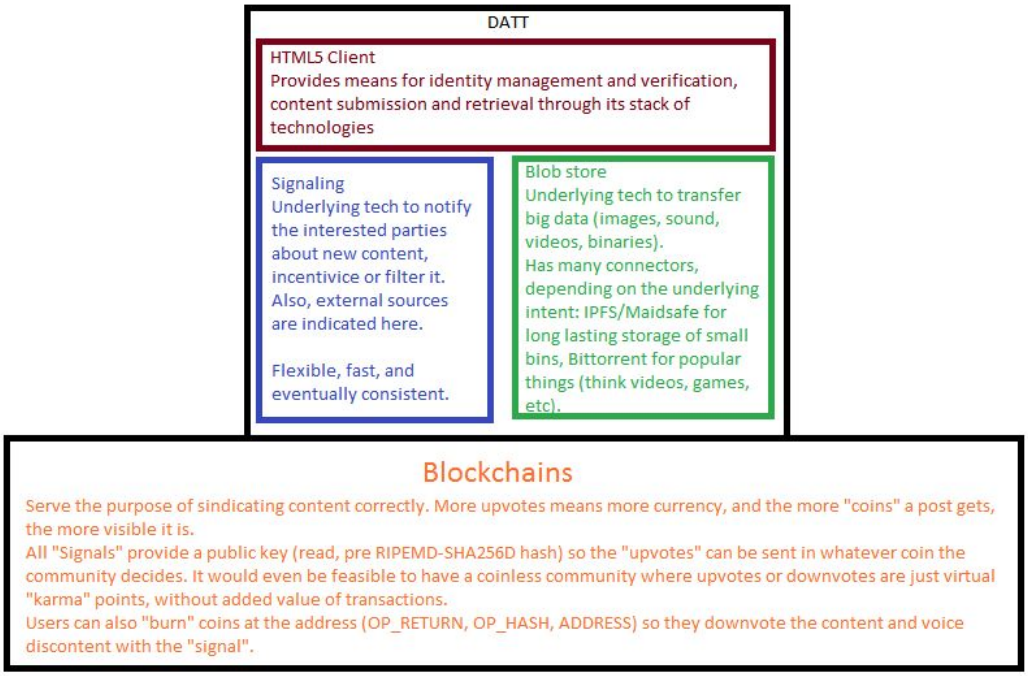
An attempt to collate all the ideas expressed on Slack so far

A quick diagram of some ideas

https://dattnetwork.slack.com/files/chiguireitor/F08EU6SR1/datt_quicky_diagram.png

A collection of Various Links from Slack(#general,#business)

- Monetization
- DATT Allocation
- Transaction Costs
- Moderators/Mods
- Downvoting
- Muttr Whitepaper
- Coalescent
- Business Case



Listings/Collectives

[Listings and Collectives were first proposed by @go1dfish here - but evolved slightly during the slack channel discussion]

By organizing communities (collectives) as organizations of listings with set logic it becomes possible to fork a community to change the moderation without losing participation or desirable curation.
[Add filtering ideas here] A filter is just saying you want to see a listing minus another listing + also list vs Filter argument

Ideas were proposed regarding tags as filters for moderation or even as subs (subreddits). Some implementation details clearly still open, like how to have different rules for subs, which likely can't be done solely with tags.

List/Tags as public keys

[Another good idea from @go1dfish discussed on slack]
Are Tags different from Listings/Collectives?

Tags can be thought of as analogous to Listings. The way I describe Listings they are typically owned by a single participant, but you could also imagine multisig controlled listings or listings with a public privkey making them open access. These listings are identified as public keys, but would have optional labels associated with them. These labels would not be unique, I could maintain my own "programming" listing separate from yours. - go1dfish

Only decrypting content when you want to read it

[yep discussed on slack leading up to here]

By separating listings of content from the actual content, clients are able to use the metadata (listings) to filter down to what actual content they want to download.

Offchain Comments similar to BitMessage

[Suggested by @chiguireitor here and reasoned here]

Content Hashing

EphemeralP2P teaches us that hashing a JSON representation of the content is very useful for both broadcasting (have & want channels) and verification that the content served is what was requested. An extension of this is to supplement the JSON with the Author's tipping address, Header image (optional - Base64 encoded), txid of the Author's posting fee, in addition to the Title & Body.

Comment Hashing

Comments exhibit less uniqueness than content (i.e. "+1" or "First post") therefore comments must both reference the Content hash, it's own comment hash, and the micropayment address of the commenter to ensure uniqueness.

go1dfish: why not just identify unique comments (and submissions) by a blockchain address, and store the content hash as a OP_RETURN?

Karma on comments

There needs to be a hub that brokers the micropayment channels specific to commenters, the primary purpose of the hub is to adjust comment Karma up/down and make the corresponding adjustment to the micropayment channel balance in favor of the commenter. Peers that request comments for a given post are likely to receive all comments from peers, therefore calls to the hub are required to determine the karma of each comment.

Wait where's the blockchain?

Ok so it's obvious that payments are happening on the blockchain (Bitcoin?), but the content hash should be included in the posting fee transaction (OP_RETURN on a small output) so there is both proof of existence and if a peer address could be included it would improve discoverability of new content.

Address management

It would be quite useful for the client/wallet to keep track of all the addresses used in authoring, peering, tipping, or commenting. BIP44 could be followed to split addresses into different 'Accounts' and this would allow for the client/wallet to summarise the total inputs and outputs by account (perhaps graphed over time for the chart lovers)

See: https://en.bitcoin.it/wiki/Deterministic_wallet

Every instance of the webapp should derive a new address - Opening the new post page in a browser tab derives a new ‘Author’ address - Viewing any existing posts in a browser tab derives a new ‘Peer’ address - Clicking to comment on a post derives a new ‘Comment’ address - Clicking to tip a post derives a new ‘Tipping’ address.

go1dfish: I don’t think we need the concept of new addresses for an ‘author’ field at all. The default state of content should be anonymous. The content is identified itself by a publickey/address. There is no need for throwaway identities when identities are a fully optional component of the system IMO.

PeerJS uses the concept of a PeerID to help the PeerServer to broker connections between nodes. We could consider the ‘Author’ or ‘Peer’ wallet address as a surrogate for PeerID.

Paul’s Original Vision *(this needs editing to reflect a broader vision not just mine or we replace it entirely)*

Content Authors are quite simply rewarded for the quality of their content by the tips they receive. A tip is an implicit upvote for that post and there is no downvoting of posts.

Peers are any clients who are currently serving a copy of the post content, thus Peers should be rewarded with micropayment channels such that they are incentivised to continue serving the content for a longer duration (this can be an exponential growth function). One option is to apportion the tips so a variable amount goes to the Author and the remainder to Peers, another option is for the Peer requesting the content to pay a small amount to the Peer serving the content. Additionally the Author may set a bounty that is paid out to peers along the same function curve.

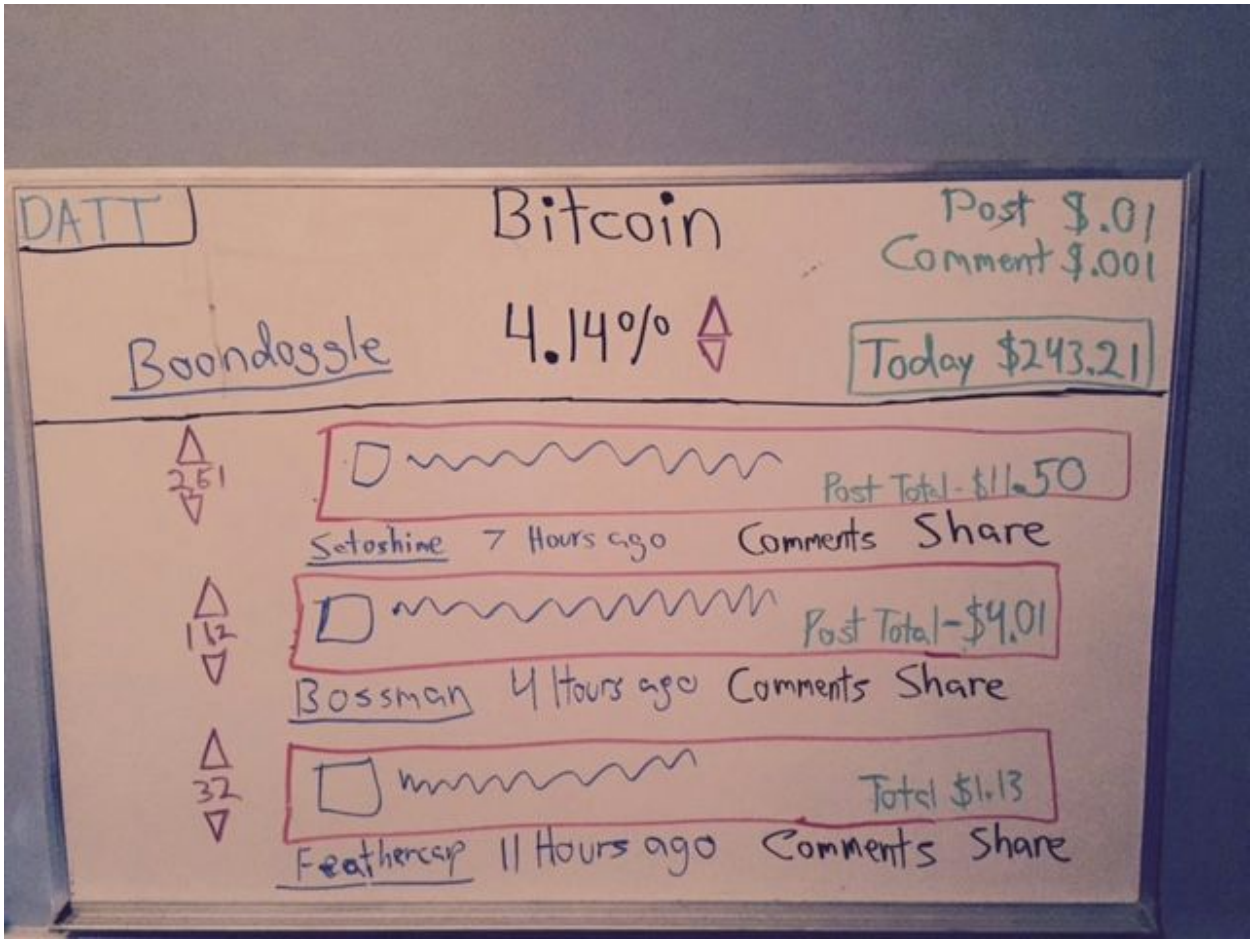
It should not be possible to comment unless the client has a positive wallet balance and the peer has been actively serving the content for a minimum length of time (e.g. 10 minutes) as this hinders spammers from spawning new peers to post repeatedly. Every comment should require a micropayment (100 mBTC) and the peer’s address is hashed along with the comment text, ready to serve to peers requesting comments for a given post. Comments could start with a base karma of 100 and this would be adjusted by up/down votes - the resulting karma between 0-100 could dictate a percentage of the amount refunded when the channel is closed.

Recap of the incentives: If you post good/popular content you are likely to get rewarded. If you serve good/popular content you are likely to get rewarded. If you write good/constructive comments you are likely to get your money back. If you troll you are wasting your time/money and are likely to fall below the karma threshold for being visible.

How does the project team get rewarded for building this platform? First and foremost a small fee should be charged for posting content. There is also the possibility of running a Super Peer node that is smart about which content to serve and when to stop serving. There is no monopoly on this and if popular we should expect to see many competing efforts to capture the maximum incentives. It is also possible that a fraction of the tips to the author are apportioned to a development fund.

Proposed Node: A Reddit like application where users pay a predefined amount directly to creators when they vote on content.

- User: Votes on posts and comments. Pays the content creator.
- Creator: Create Posts and Comments. Gets paid by Users and pays Mods.
- Mod - Controller of the channel. Gets paid by Creators and pays Nodes.
- Node - Gets paid by Mods. Pays Datt.
- DATT - Gets paid by Nodes. Builds protocol.



1. Users pay a predefined amount to vote on a post and another predefined amount to vote on a comment
2. Creators receive the majority of the money from users, but pay mods a fee
3. Mods curate/moderate and receive a % of fees paid to creators in exchange for that service
4. The amount that creators receive for votes on posts and comments is defined by the mod
5. Users upvote/downvote the % fee the Mod should take
6. The % fee that the Node takes is determined by the mods

7. The % fee that Datt takes is determined by the Nodes
8. Nodes that display content hosted by other Nodes give 100% of fees to the hosting Node
9. Any Node can choose to share or block any other node
10. All fees made by the content creator are displayed within the post
11. All fees made by the Mod are displayed at the top of their channel

Why is this better than the current system at Reddit?

- Mods are financially motivated to increase the # of votes. They will focus on content curation.
- Content creators have an incentive to post content here instead of on traditional media platforms where they only receive ad revenue.
- Nodes are financially motivated to provide Mod friendly policies around censorship, create better tools for Mods to curate, and provide good user experiences.
- Legal liability arising from receiving a % necessitates removing copy written content and ensuring that the right people are paid.
- Legal liability arising from receiving a % necessitates removing criminal content.

Pledges:

Mod Oath of Office -
I swear that I will not take any money as side deals that may arise from my role as your moderator without completely and wholly disclosing it to the community in advance of completing the transaction. I promise to resign if I intentionally violate this rule.

Creator Mod Oath of Office -
I promise to be as awesome as possible.

Node Oath of Office -
I promise to facilitate and not interfere with creators, moderators and users. This includes reporting those that break the law as well as removing those that violate the community's ethical standards. We promise to resign if we violate this rule (signed by every member of the company).

https://docs.google.com/spreadsheets/d/16vtDV3Ot4lrr2twpUS8Sqgg4OrJ2KlrnQwsuppy56sI/edit#gid=0