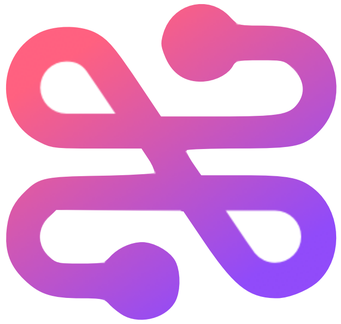


fission

BUILDING WEB NATIVE APPS

IPFS COMMUNITY CALL MARCH 2020

fission

WHAT IF FRONT END CODE WAS ALL YOU NEEDED?

- Front end is never going away
- Browsers keep getting **more powerful**
- New public infrastructure
- Rise of edge computing

fission

WEB APPS ARE HARD

- Multiple specialities = big teams & more coordination
  - Front End
  - Back End
  - DevOps

- Lots of boilerplate
- Data security and access control are hard
- Deployment in 2020 is super complex

fission

WEB NATIVE APPS

- Integrated framework for self-contained “web native” apps
- Hosted services for a batteries included experience
- Convenience, deployment pipelines, “it just works”
- Features for Developers
  - User Accounts with Data Privacy
  - Passwordless Login & Authentication
  - Modern App hosting, DNS, CDN
  - Access Control & Encryption

fission

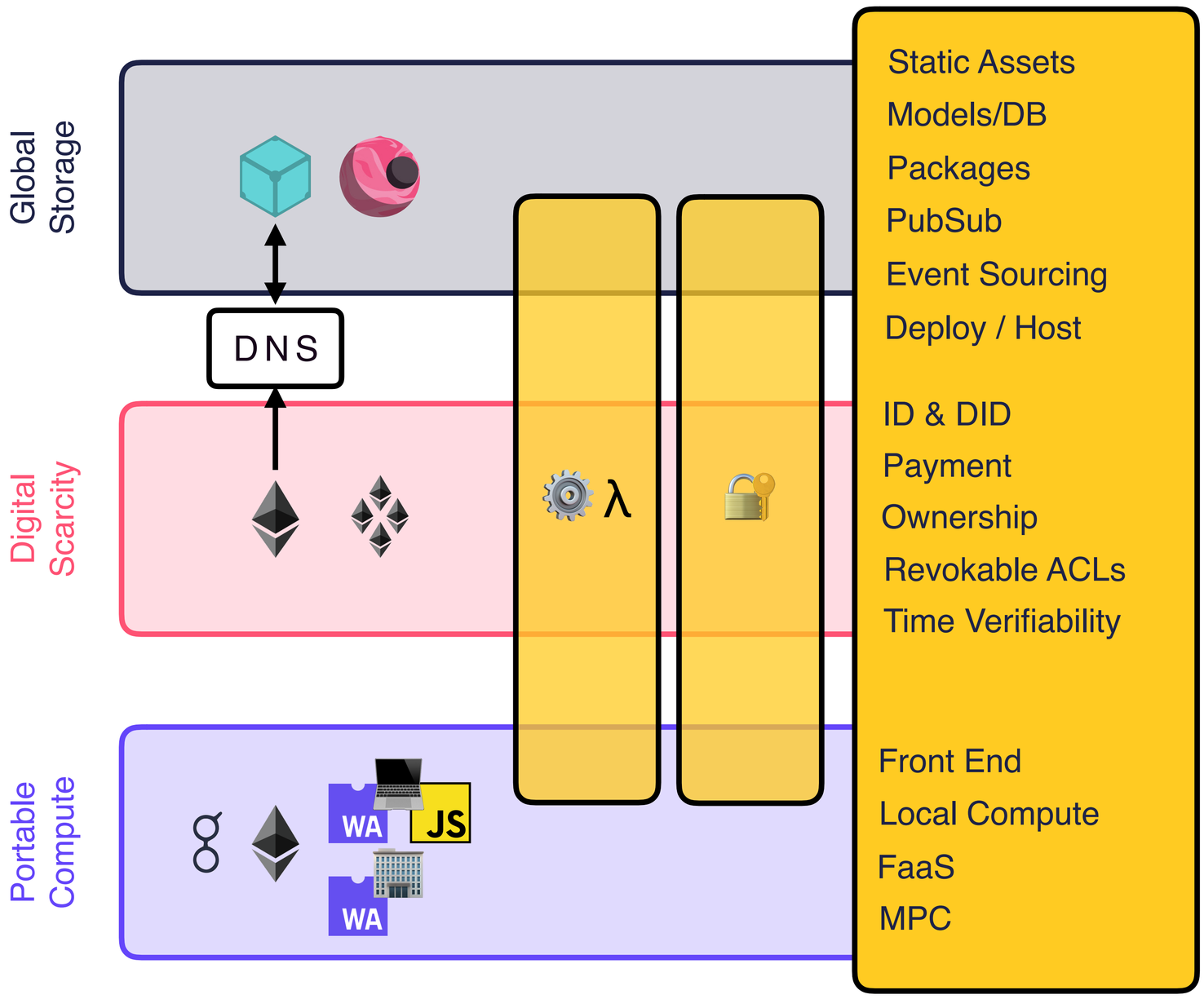
Like native apps for the web:
self contained, user-controlled data, client-side compute, etc.

	iOS	+
Compute		WA
Identity		webauthn
Storage		IPFS

fission

WEB NATIVE

- You should be able to build an entire app while offline on a plane, create the first user account locally, and have it be “deployed” by connecting to the Internet when you land
- Should work in all browsers, including mobile, without plugins

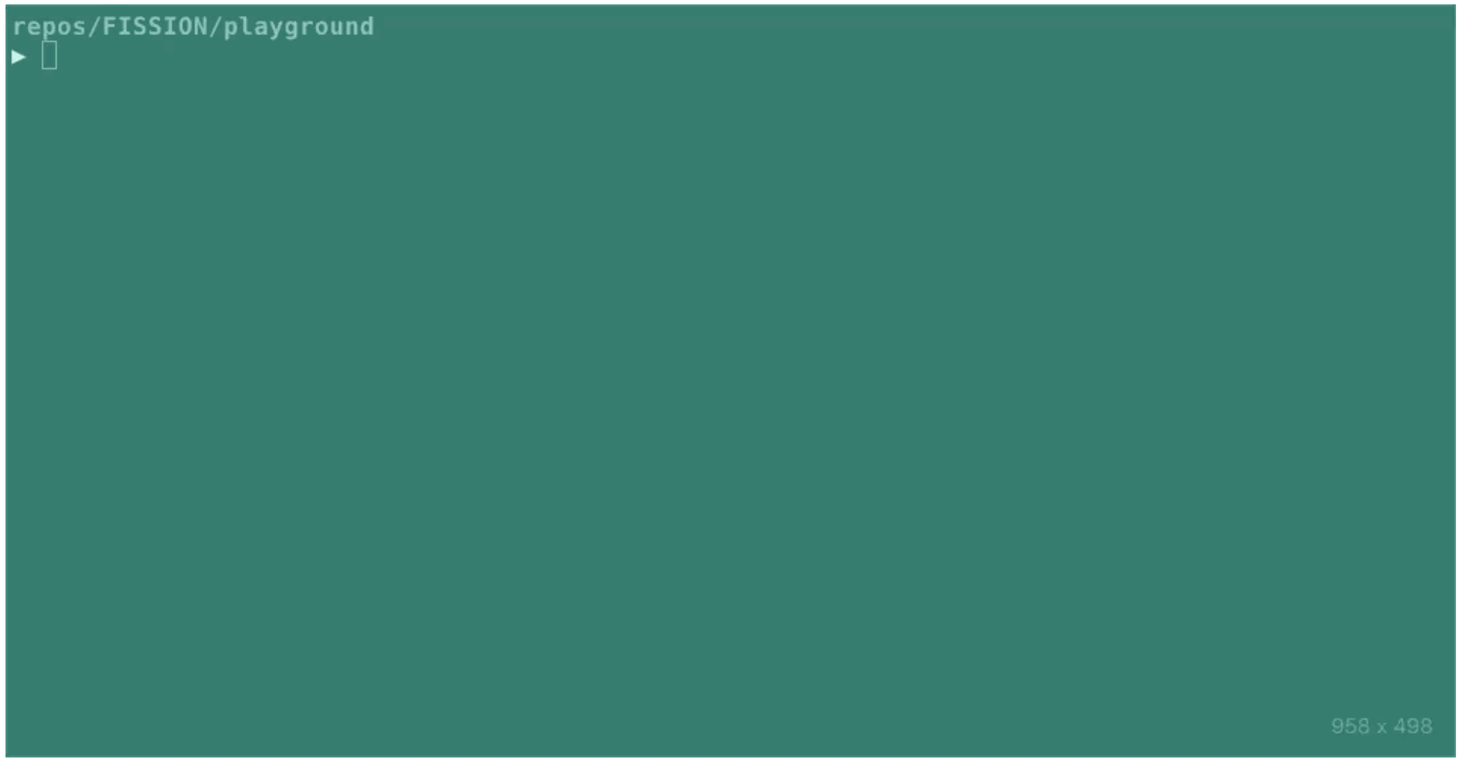


fission

BUILT SO FAR

- Web API
  - RESTful API for IPFS
  - Heroku Add-on
  - DNS Automation
- CLI
  - Devs install IPFS locally
  - Signup, login, can all be done in the CLI
  - `fission up`

Go try it! guide.fission.codes



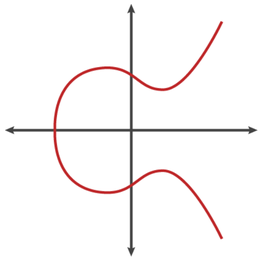
fission

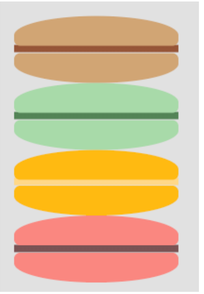
NEXT UP

- Public/Private Key based accounts
- Accounts can add Apps, Apps can have Domains — APP.fission.app
- Encryption
- Web Login
- Fission File System

fission

IDENTITY





- Built on public-private key cryptography
- WebAuthN now widely supported in browser (including mobile) means passwordless login
- W3C is standardizing Decentralized IDs and Verifiable Credentials
- JWT Auth and macaroons
- More: whitepaper.fission.codes/identity

(we probably shouldn't call this identity, rather authorization + authentication)

fission

FISSION FILE SYSTEM

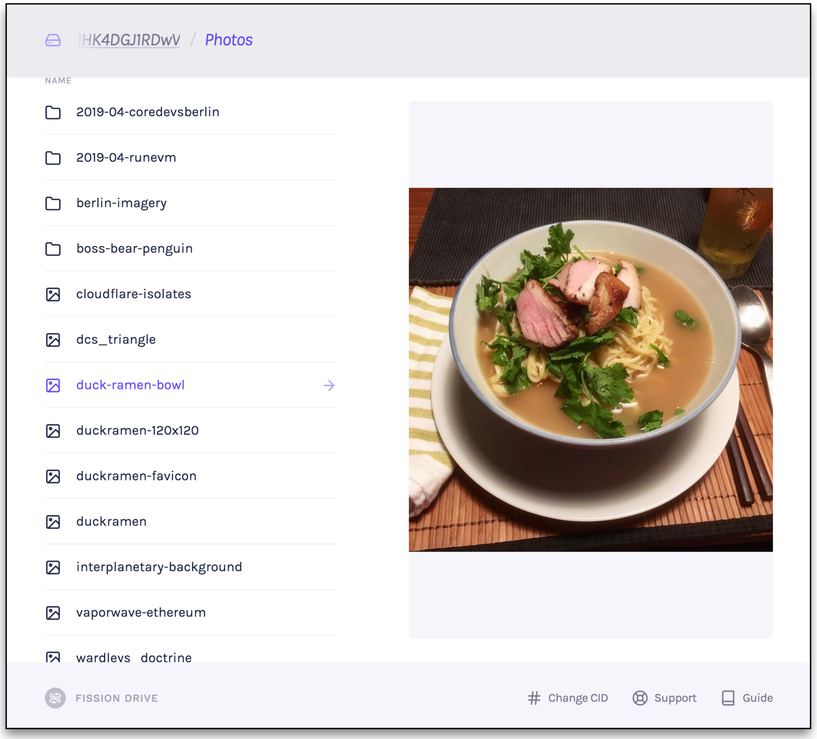
- Store the current DAG hash in DNS, eg. boris.fission.name
- Default user file tree, encrypted except for Public folder
  - Docs
  - Images
  - Music
  - Apps
  - Shared
  - Public

fission

FISSION DRIVE

Go try it! drive.fission.codes

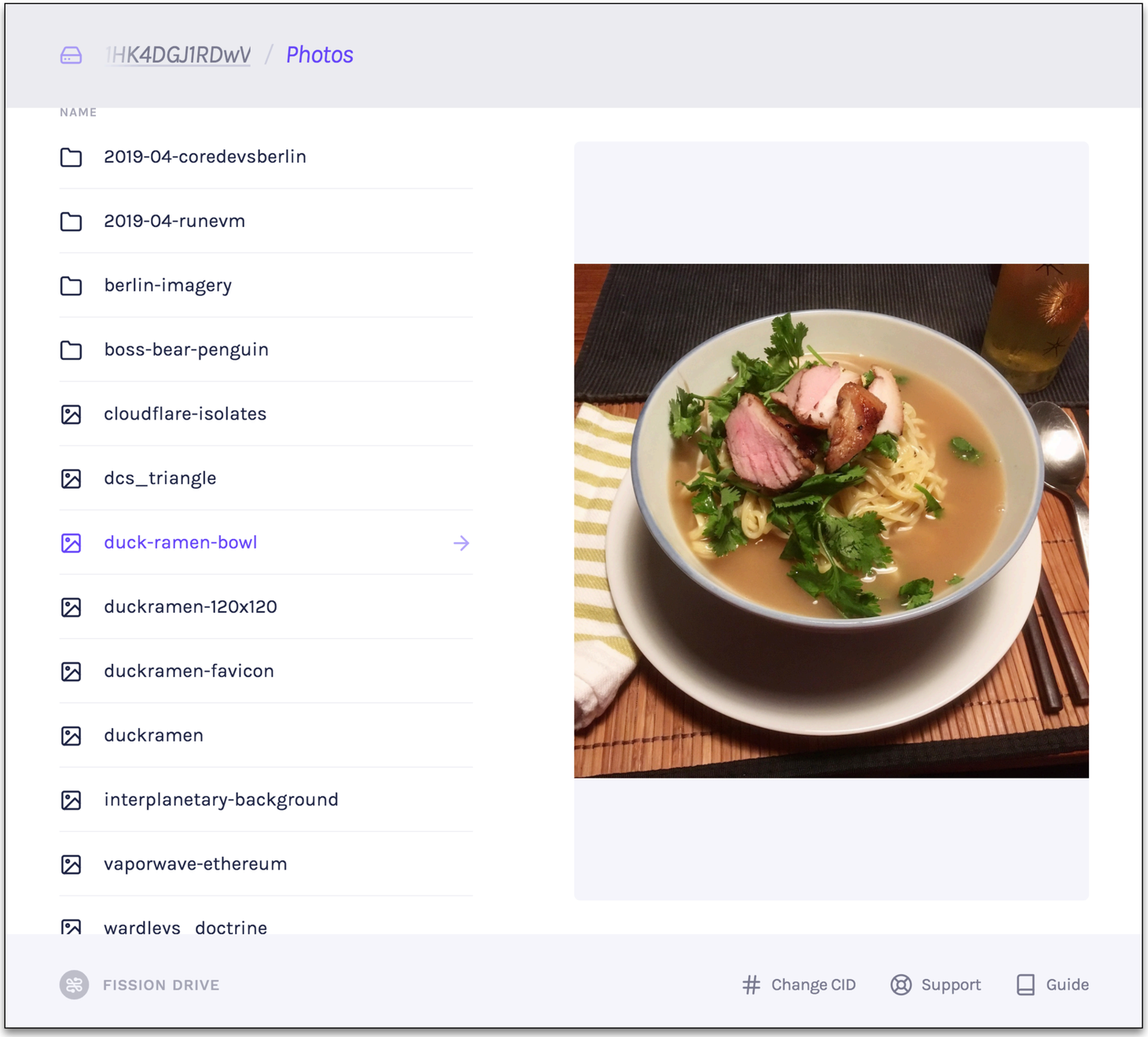
- **Our own app for end users:** early adopters and developers
- **Documents:** files, screenshots, photos, music, etc. — dotfiles?
- **App data:** per-app storage under user control — add apps to your Drive, give them permission to access your files, all client-side



fission

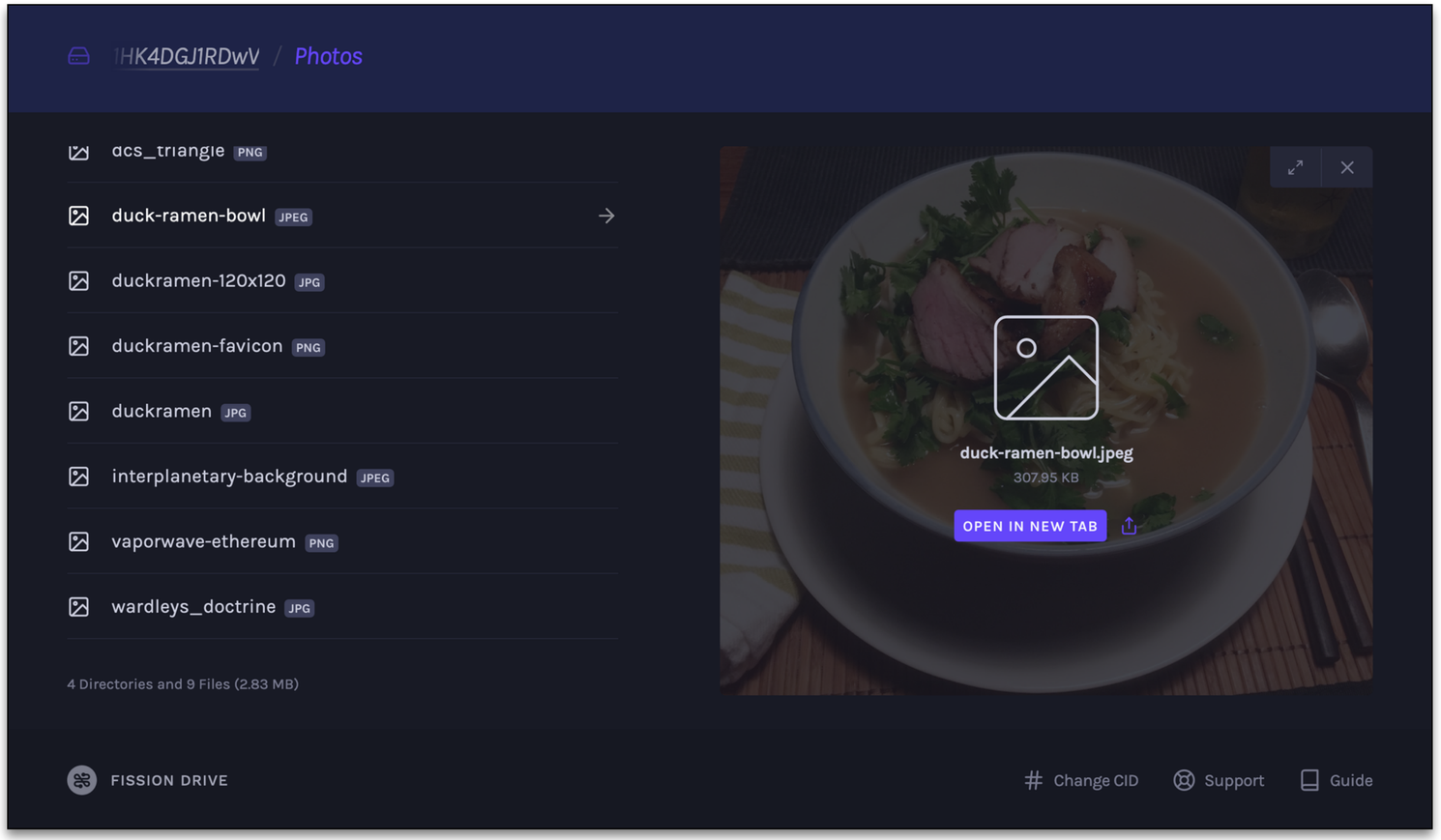
FISSION DRIVE

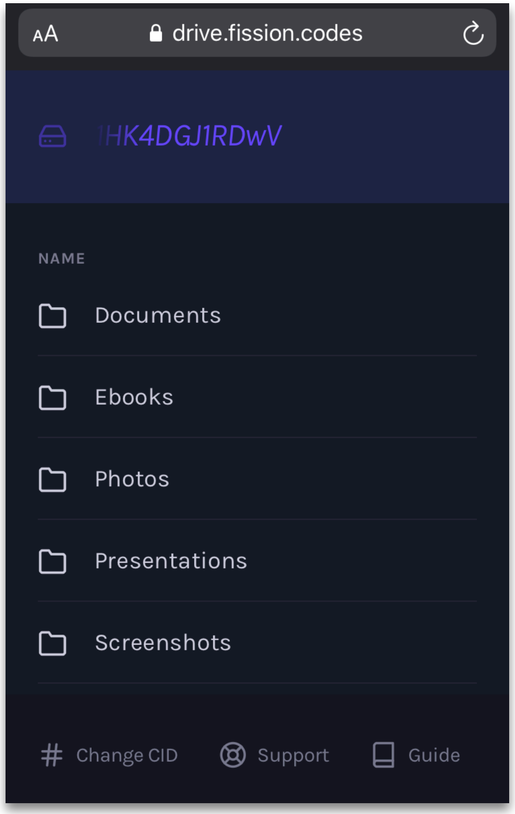
- Preview release works with any public IPFS files
- Coming:
  - Private files
  - Password-less login



fission

FISSION DRIVE





fission

CORE IPFS

- fission-suite/get-ipfs:
  - npm package to load an IPFS instance into a web-page
  - Like ipfs-shipyard/ipfs-provider, but we use ES6 syntax (and built it around the same time!)
- fission-suite/ipfs-haskell
  - Haskell package to work with local IPFS node or remote through http interface
- go-ipfs feature: “ipfs ignore”
  - We currently manage this in the Fission CLI directly (e.g. *.env*, *.ssh*, *.fission.yaml* are default ignores)
  - PR in progress to hopefully get this added upstream

fission

INTEGRATIONS

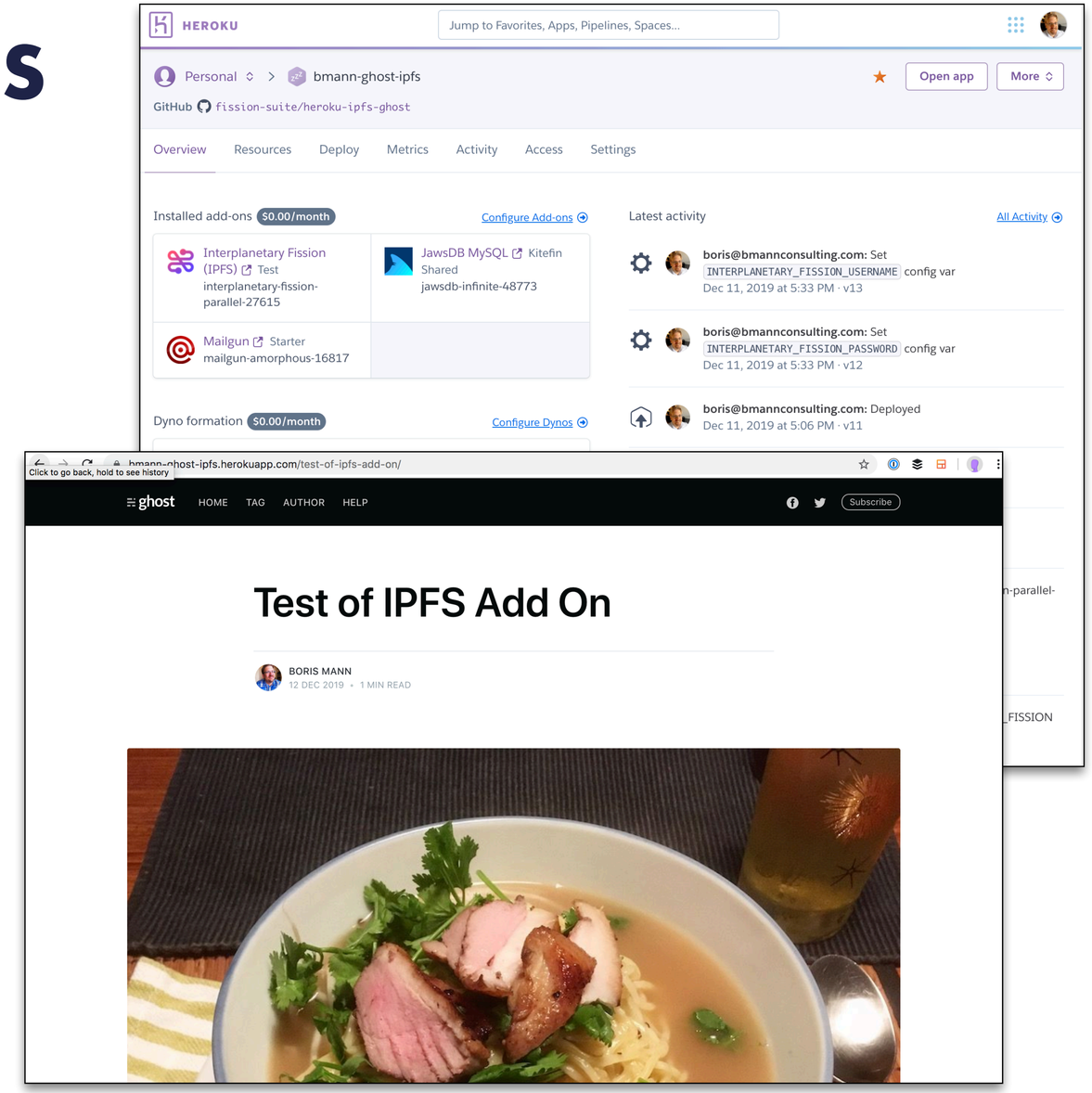
- Work with existing Web2 apps
- Provide many ways to “ingest” files into IPFS
- Make IPFS files re-usable
  - Cached everywhere
  - Sync
  - Manage on desktop

fission

HEROKU + GHOST + IPFS

- One click Deploy to Heroku: fission-suite/heroku-ipfs-ghost
- Ghost Storage Adapter: fission-suite/ghost-storage-adapter-ipfs

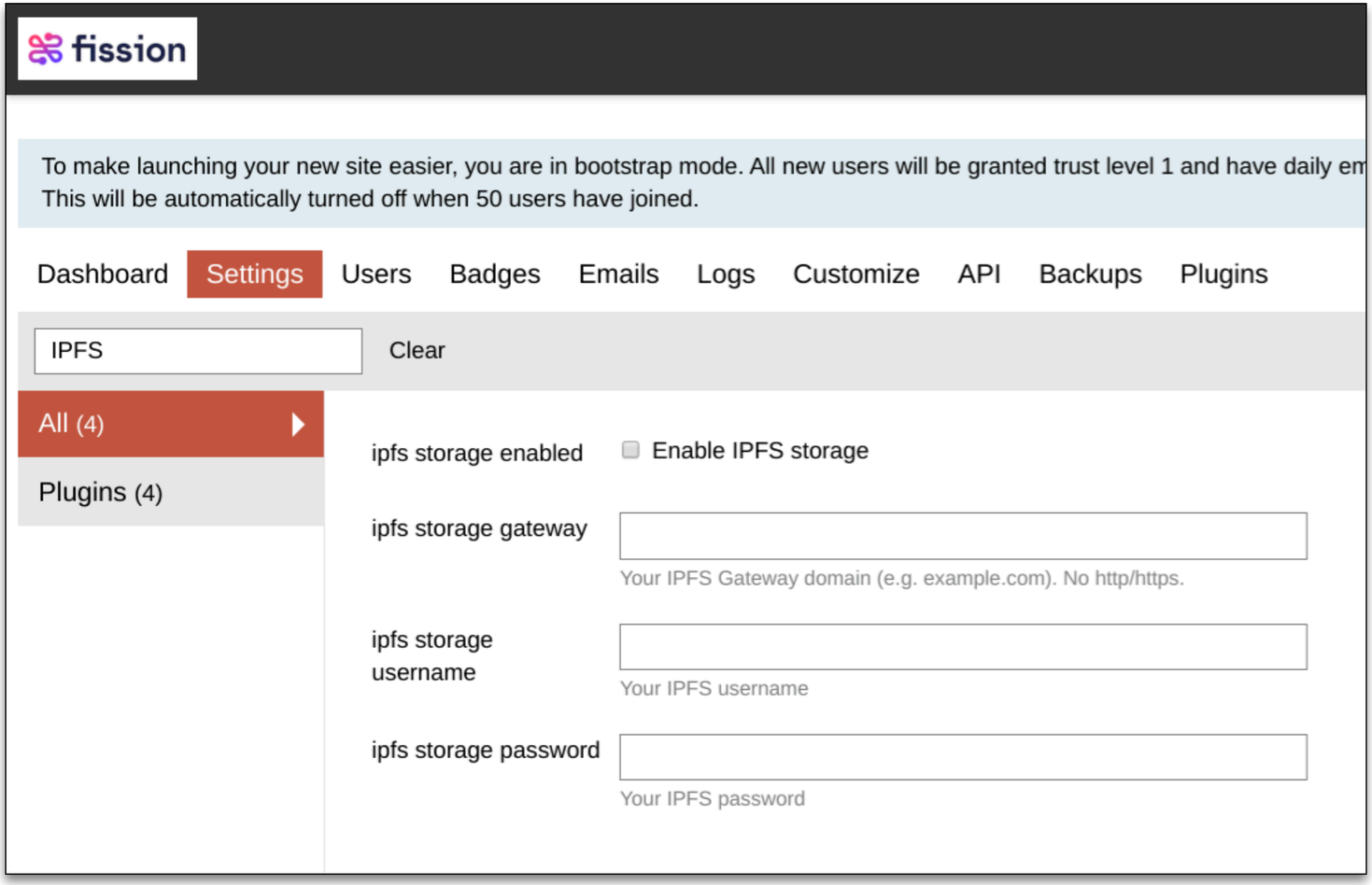
- Uses Fission's Heroku add-on

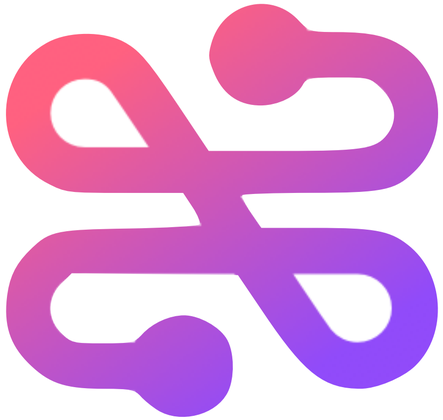


fission

DISCOURSE

- Discourse plugin: fission-suite/discourse-ipfs-file-store





fission

THANK YOU!

Github: fission-suite

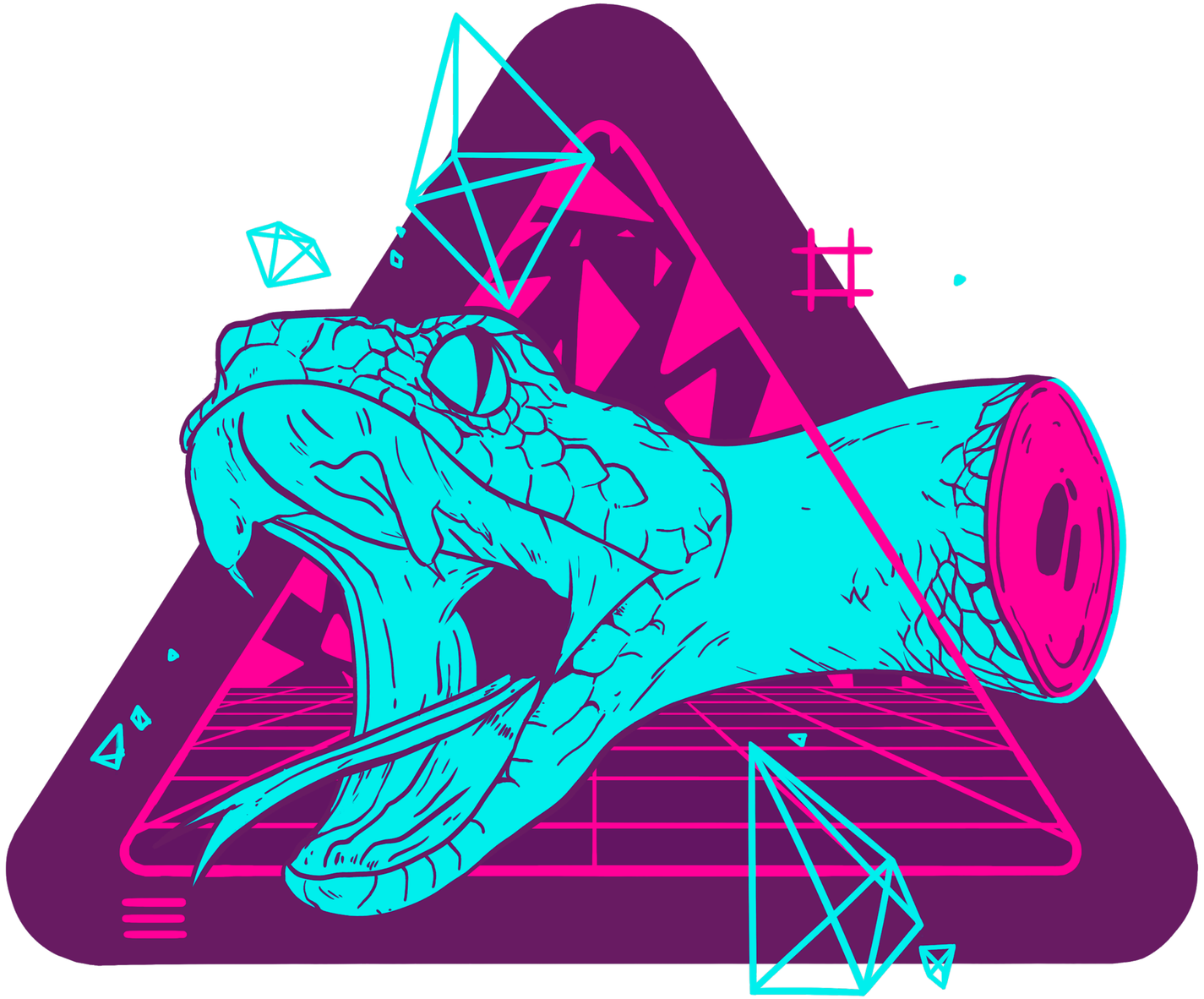
Install Guide & Docs: guide.fission.codes

Technical Whitepaper: whitepaper.fission.codes 🚧👷✍️

Discourse forum: talk.fission.codes

Discord chat: fission.codes/discourse

Twitter: @fissioncodes, @expede, @bmann, @icidasset, @dholms on Github



GET SWAG!!!

swag.fission.codes