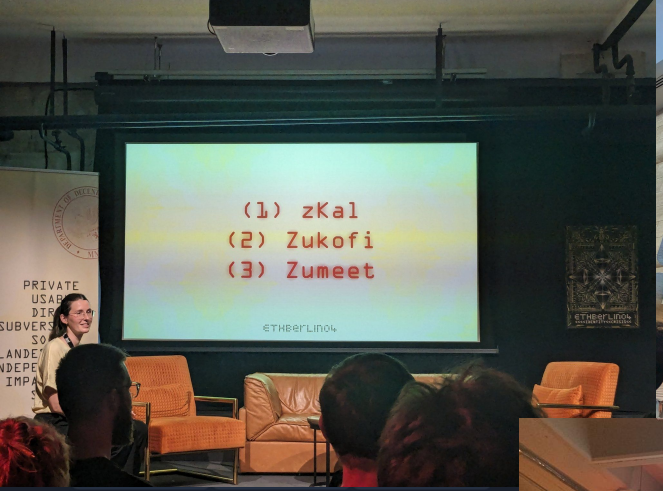
Zupass: identity and credentials

Beyond proof of personhood

Rob Knight

Programmer, Zupass team

What is Zupass?



ZuRat admin
ANONYMOUS WHALE

Frogs are yummy. Now I want to eat some frogs.

submitted 09/11/2023, 22:19:24

👍 ❤️ 2

Zuum

Connect Strava to see your map

Connect with STRAVA

Or connect via browser

You have until **June 30th** to explore Healdsburg.

Get at least **10 km** to win a 🎁 from HRC!

Announcing: Zuzalu QF Grants Program on Grants Stack

Grants Stack

Zupass is...

- How you checked in at Devcon
- How you collected frogs and claimed frog hats
- How you voted for the hackathon winners at ETH Berlin
- How you proved you attended Zuzalu-ecosystem events in Gitcoin

Zupass also helped you to:

- Ask speaker questions and give feedback for talks
- Post anon messages at Zuzalu, ZuConnect, and Devconnect
- Provide anonymous clinical data in a decentralized trial at ZuConnect
- Prove your Strava history at Edge Esmeralda

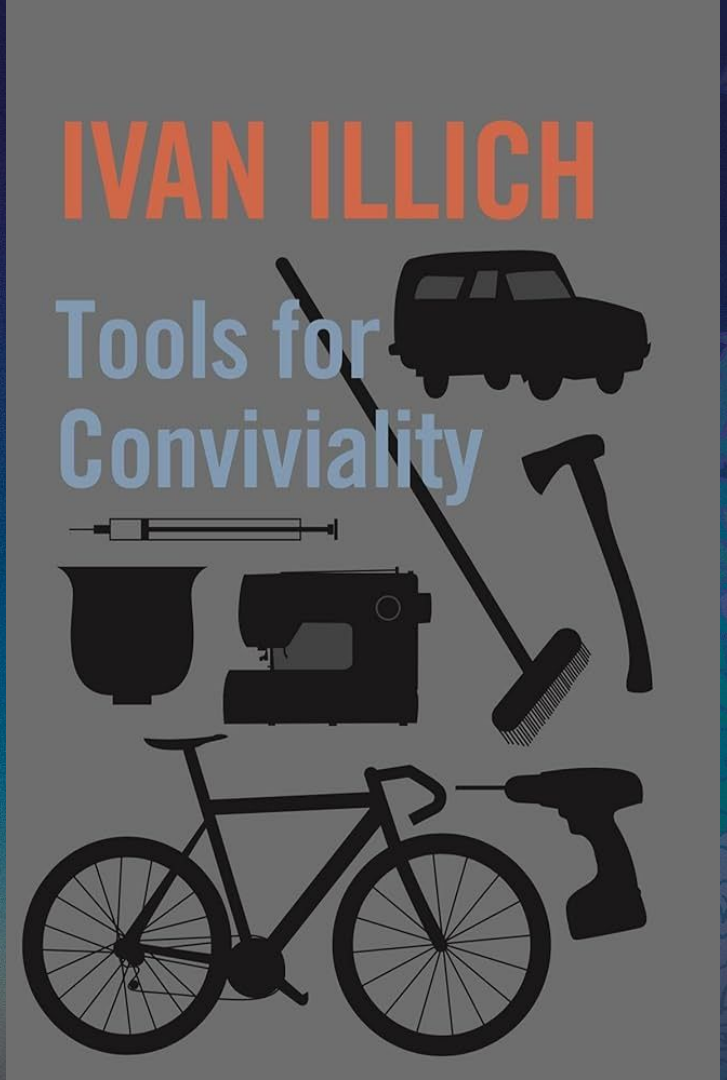
A philosophical aside on identity, order, and community

Conviviality

Literally “living together”

Meaning in English is also conveys “friendliness, agreeability”

Convivial technology enables friendly interactions within communities



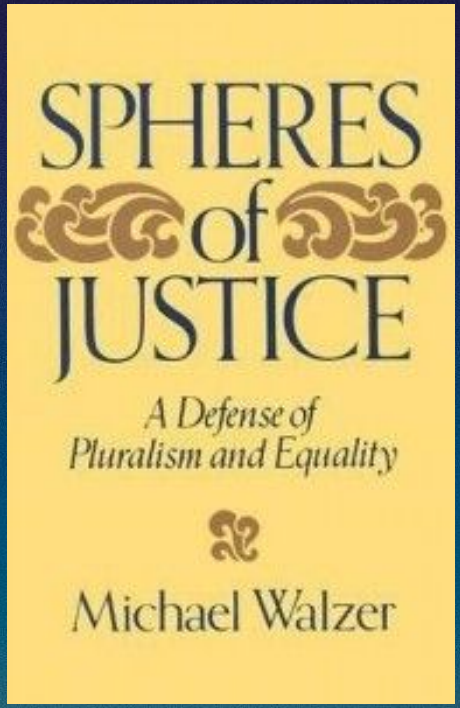
Who am I? Who are we?

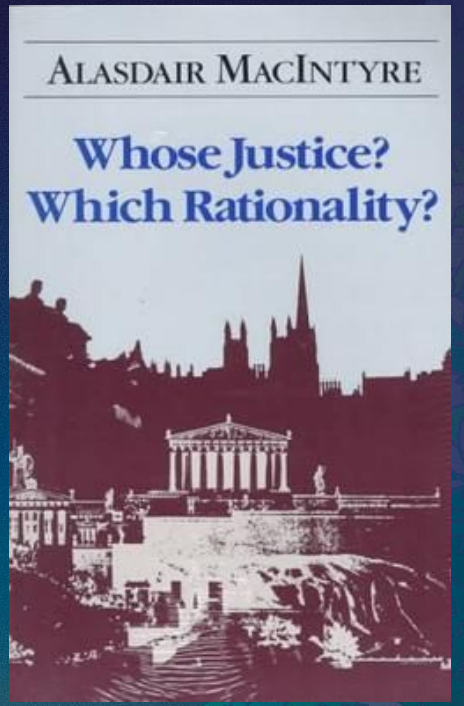
It depends on who is asking!

I am a different person to my friends, my colleagues, my family, the government.

Decentralization isn't just about data, it's about *structure* and *logic*.

Different traditions and communities have *incommensurably different* ideas.





Spheres of Justice by Michael Walzer

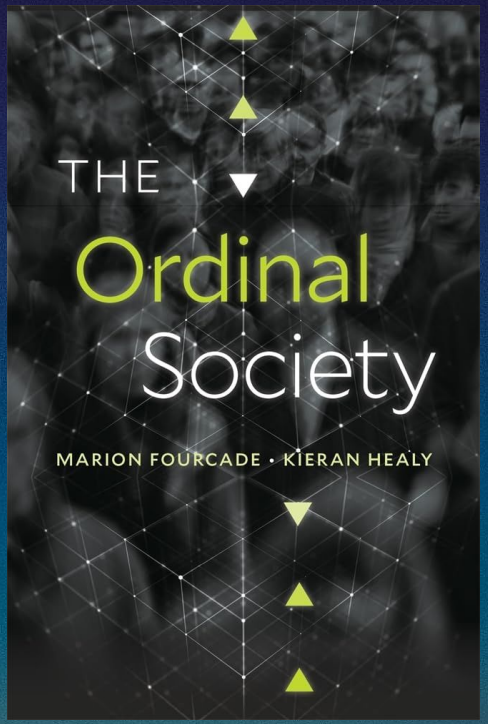
Whose Justice? Which Rationality? By Alasdair Macintyre

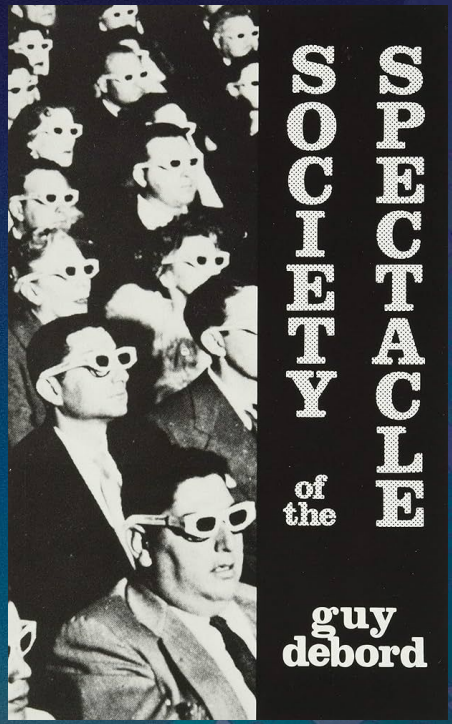
Order-givers and order-takers

People at the top give orders, and make order.

People at the bottom take orders and receive order.

Computers work by imposing order on messy reality.





Society of the Spectacle by Guy Debord

The Ordinal Society by Marion Fourcade and Kieran Healy

POD, GPC, and Z API

POD, GPC, and Z API

POD = Provable Object Data (sometimes the ‘P’ stands for ‘Portable’)

GPC = General Purpose Circuits

Z API = Embed Zupass in your app

PODs let you create your own structures

PODs let you define arbitrary data structures.

You can share these structures with other people, letting them make the same kind of POD.

PODs can be freely shared without any central API.

```
export const FrogSpec = p.entries({
  pod_type: {
    type: "string",
    isMemberOf: [{ type: "string", value:
POD_TYPE_FROGCRYPTO_FROG }],
  },
  name: { type: "string" },
  description: { type: "string" },
  imageUrl: { type: "string" },
  frogId: { type: "int", inRange: { min: 0n,
max: POD_INT_MAX } },
  biome: { type: "int", inRange:
enumToRange(Biome) },
  rarity: { type: "int", inRange:
enumToRange(Rarity) },
  temperament: { type: "int", inRange:
enumToRange(Temperament) },
  jump: { type: "int" },
  speed: { type: "int" },
  intelligence: { type: "int" },
  beauty: { type: "int" },
  timestampSigned: { type: "int" },
  owner: { type: "cryptographic", isOwnerID:
true },
  ownerPubKey: { type: "optional", innerType:
{ type: "eddsa_pubkey" } },
});
```

GPCs let you create your own logic

GPCs let you build zero-knowledge proof from configurable building-blocks.

Ask for the proof that you need, not what someone else thinks you need.

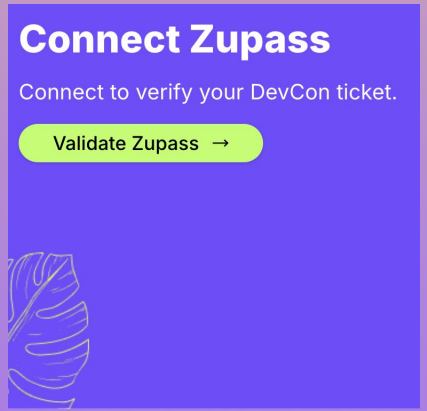
GPCs can take in any kind of POD, or multiple PODs.

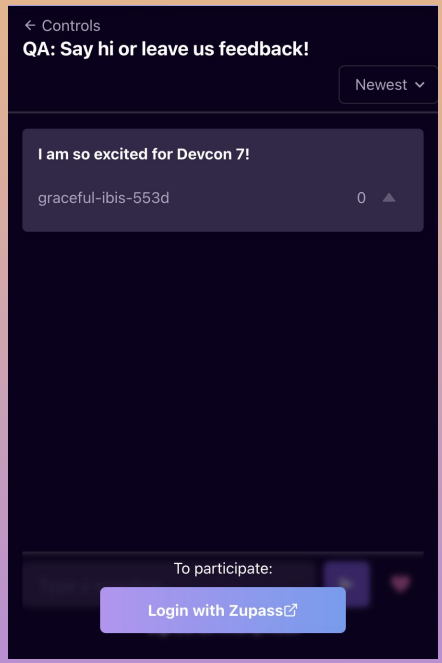
```
const result = await z.gpc.prove({
        request: {
          pods: {
            FROGCRYPTO: {
              pod: {
                entries:
entriesToProve,
              },
              revealed: {
                beauty: true,
                jump: true,
                speed: true,
                forgId: true,
                name: true,
                biome: true,
                owner: true,
                intelligence: true,
              },
            },
          },
        },
      });
```

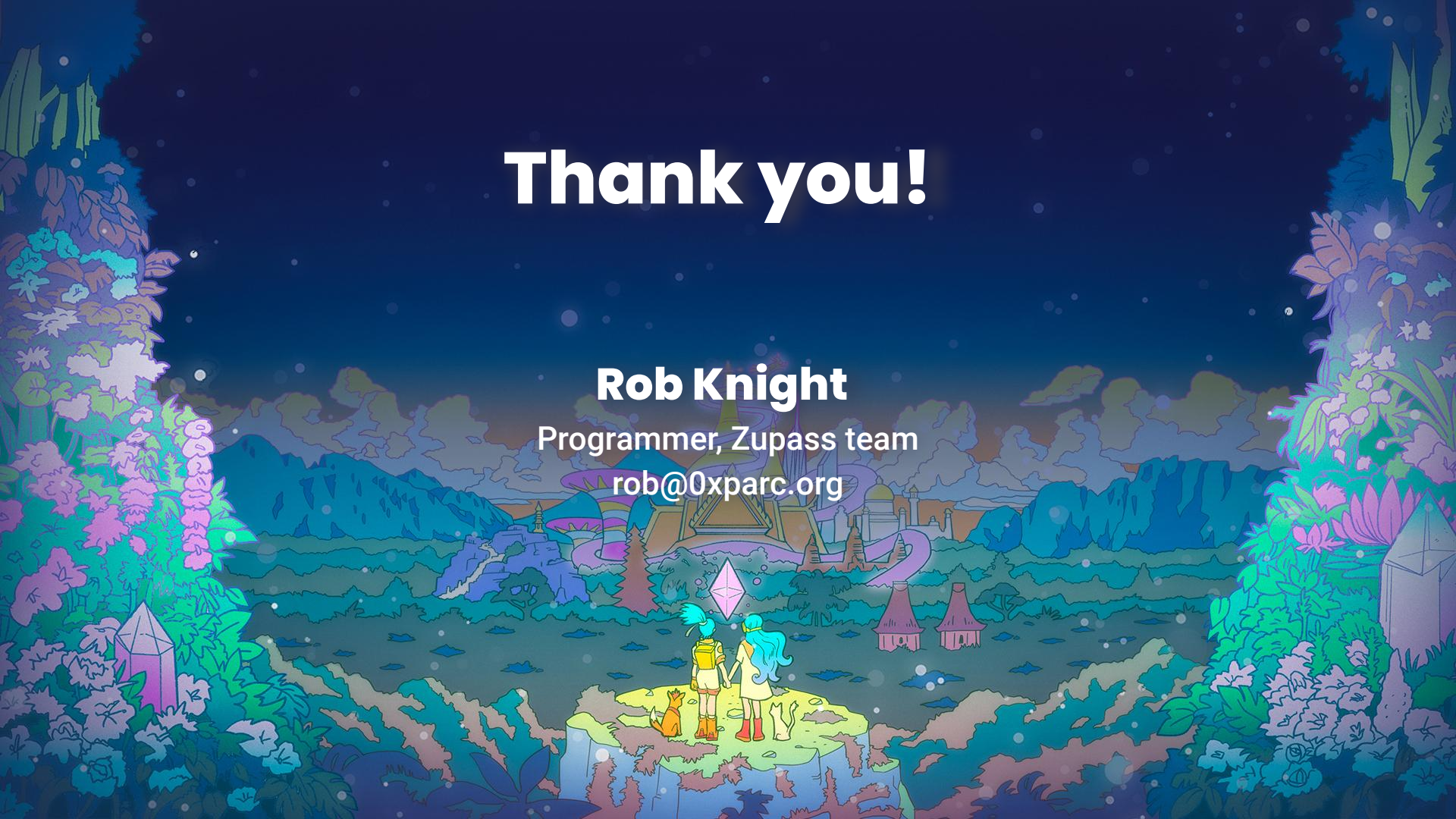
Z API: build your own app

The Z API provides an integration with **Zupass**, which gives your app:

- a pre-built POD store with end-to-end encrypted synchronization
-
- management of cryptographic keys and identity
-
- UI primitives for zero-knowledge proofs






Thank you!
Rob Knight
Programmer, Zupass team
rob@0xparc.org

POD, GPC, and Z API

POD = Provable Object Data (sometimes the ‘P’ stands for ‘Portable’)

GPC = General Purpose Circuits

Z API = Embed Zupass in your app