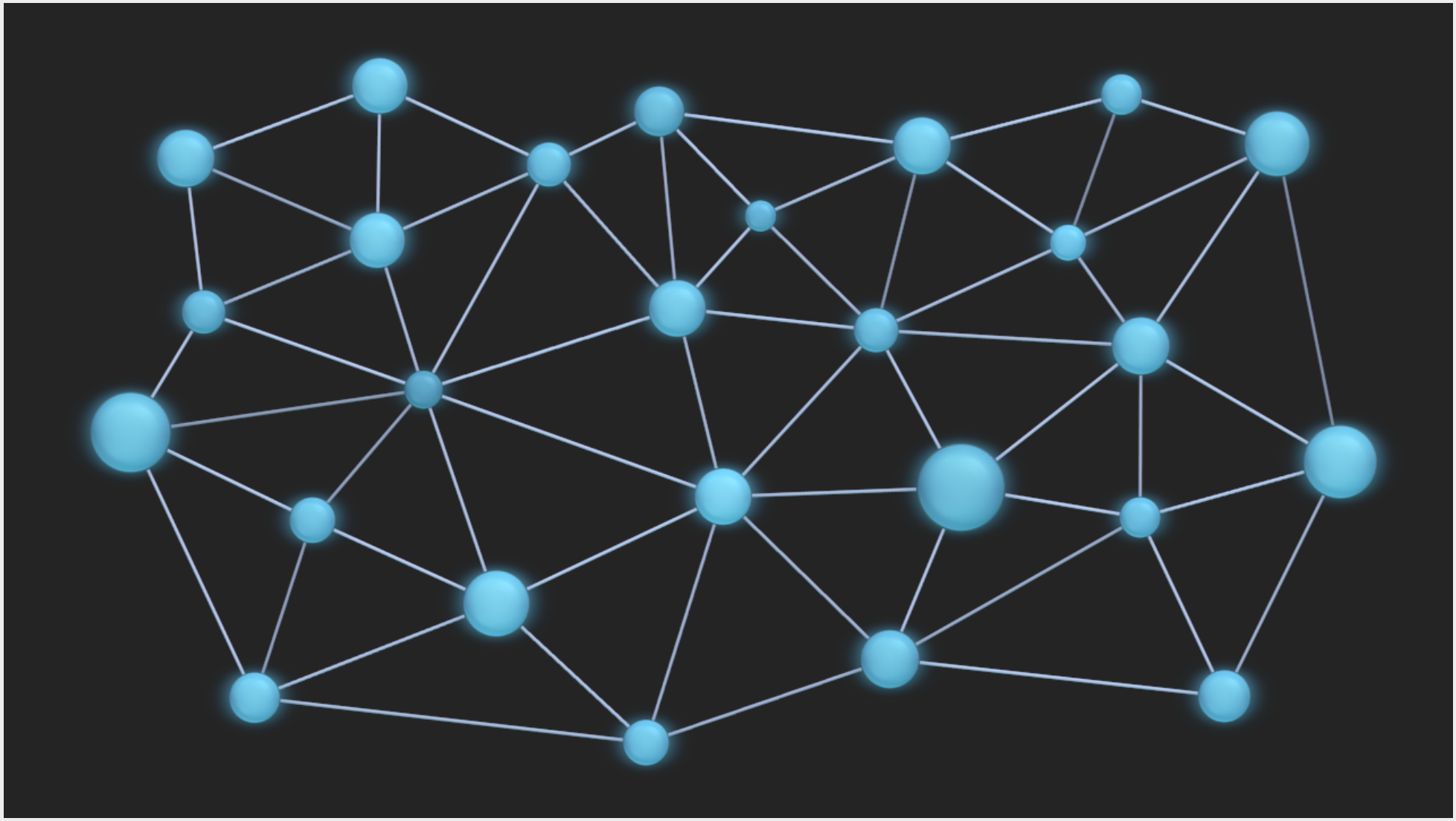
Goblins and Spritely

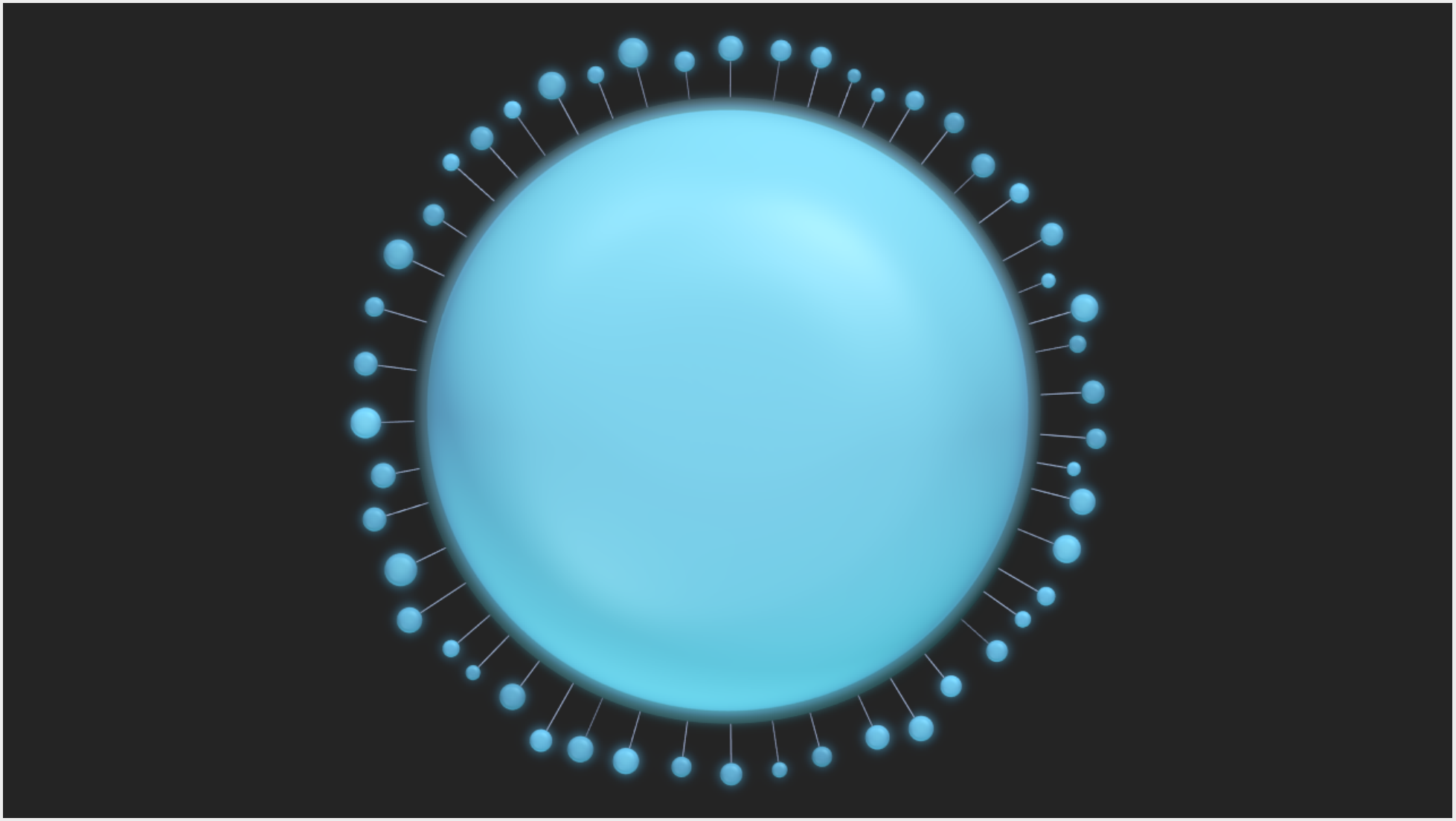
From the actor model to Distributed Virtual Worlds

By Christopher Lemmer Webber

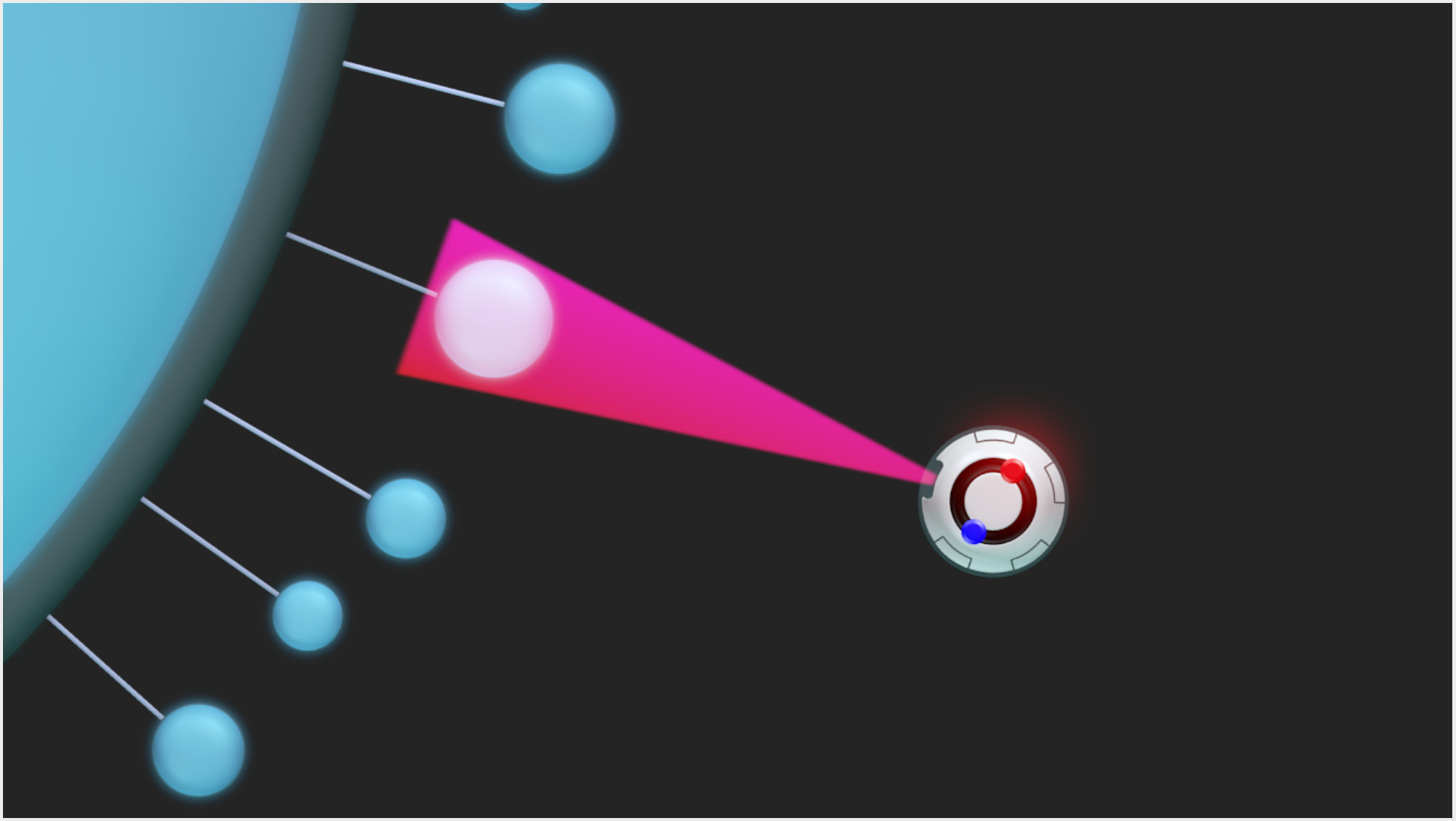
The web we want



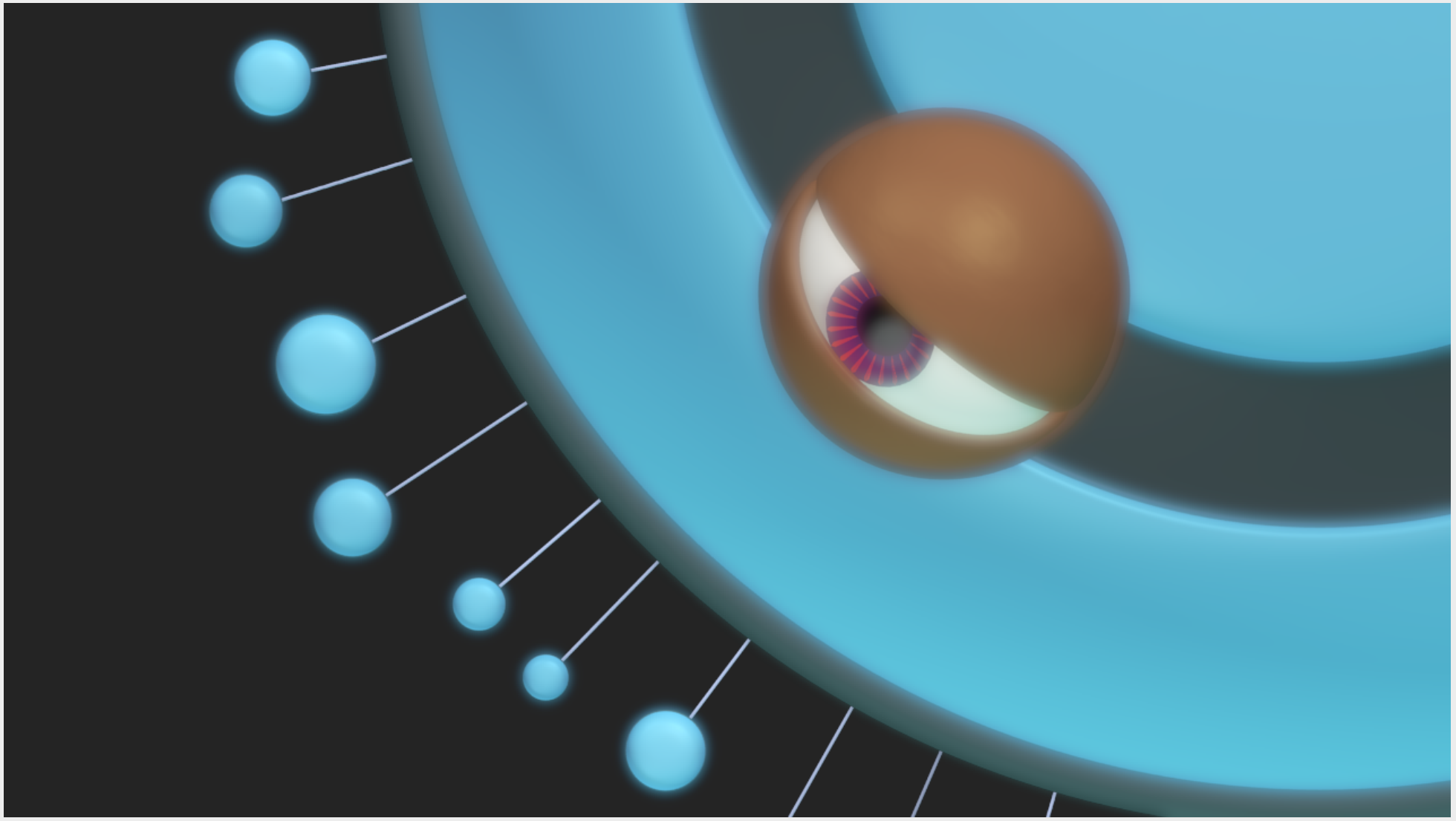
The web we got



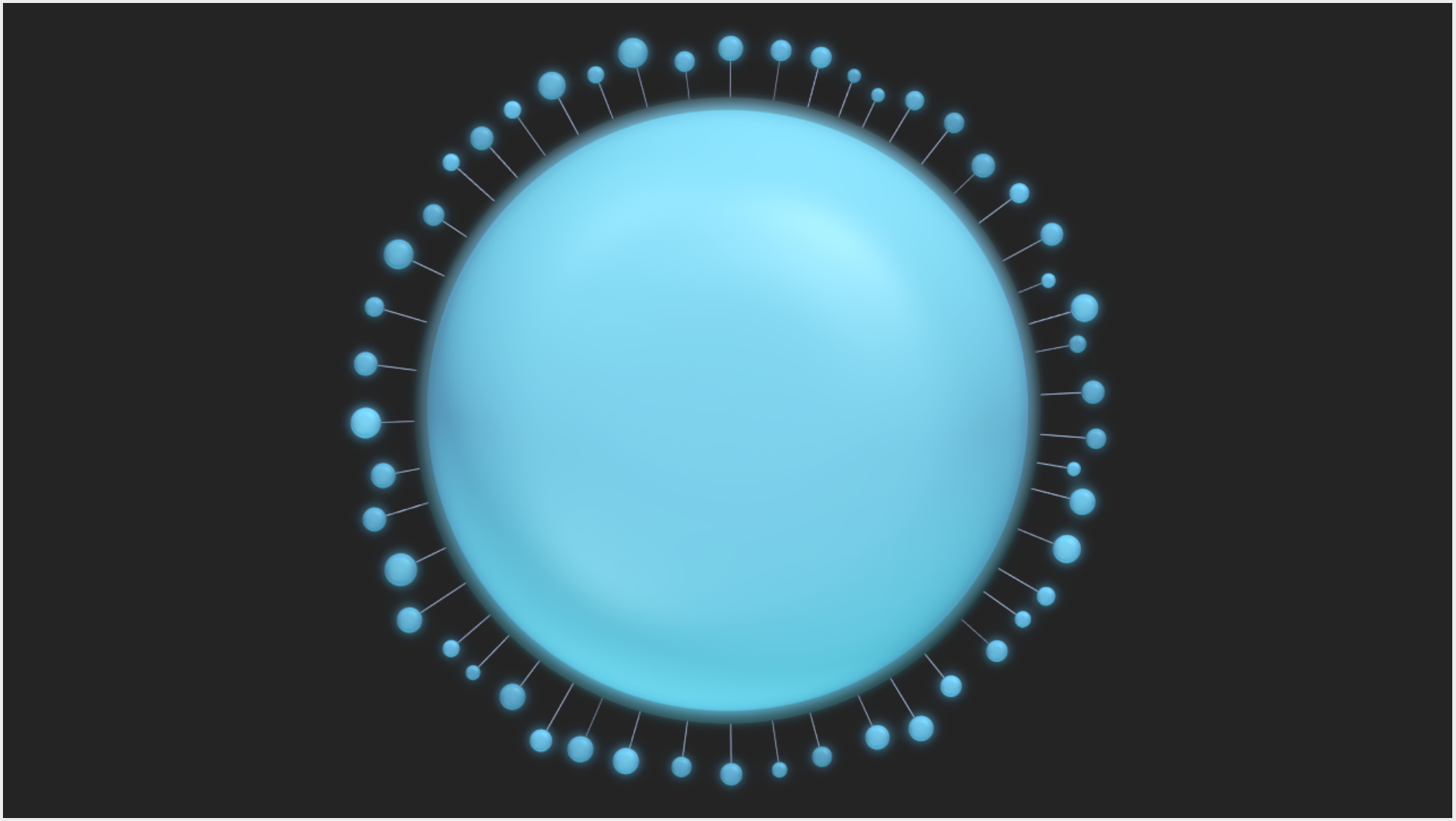
The web we got



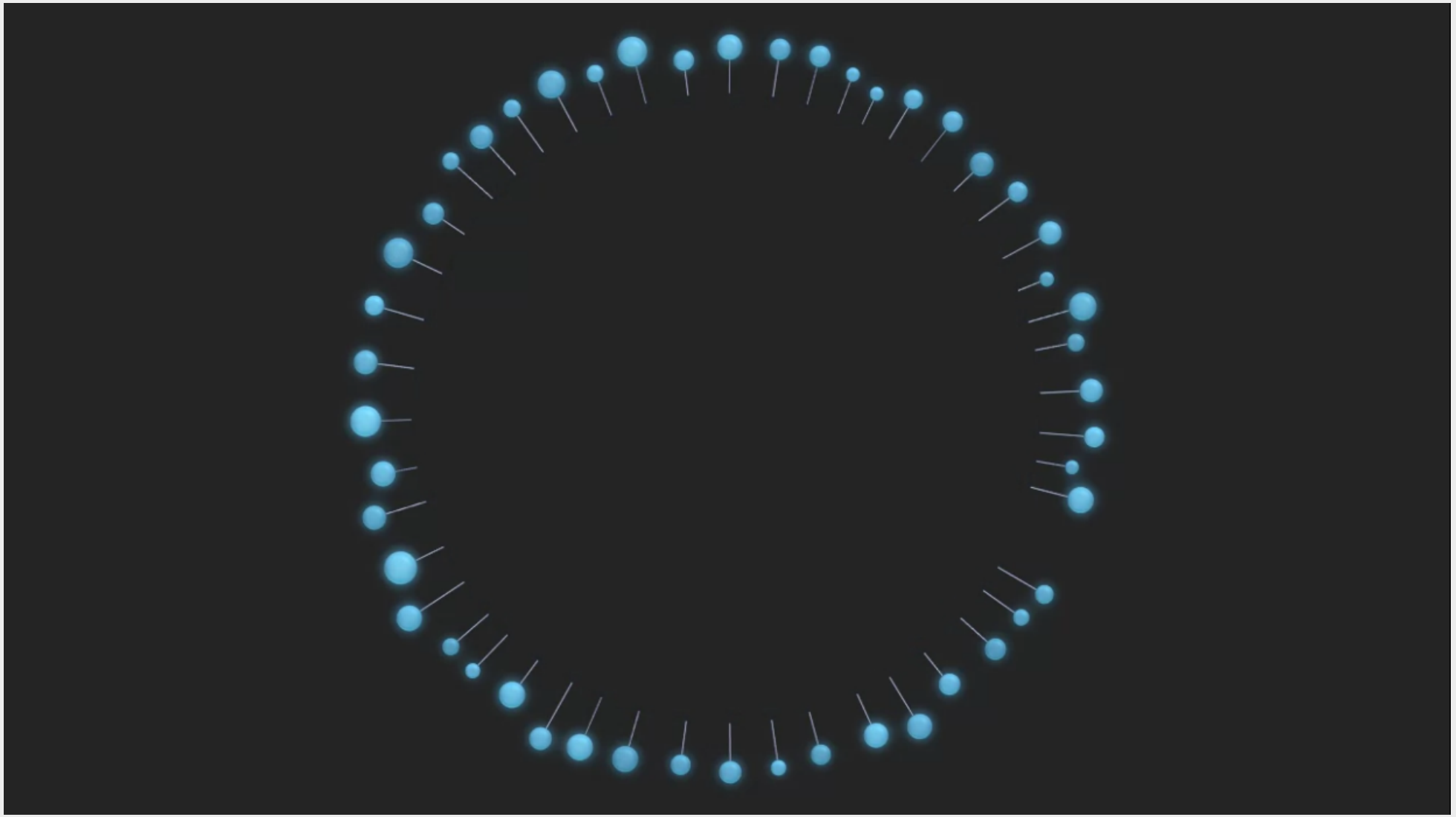
The web we got



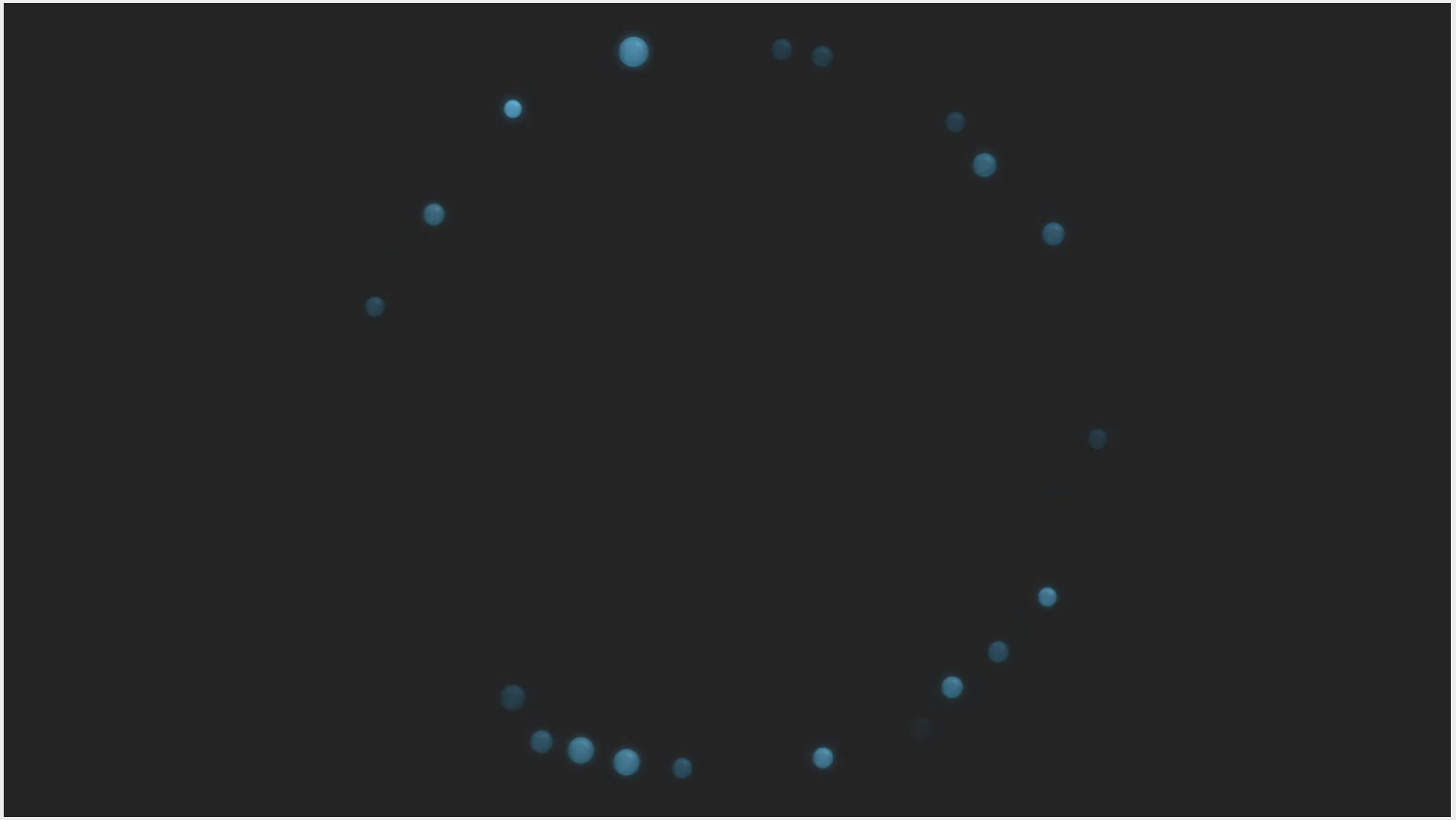
The web we got



The web we got

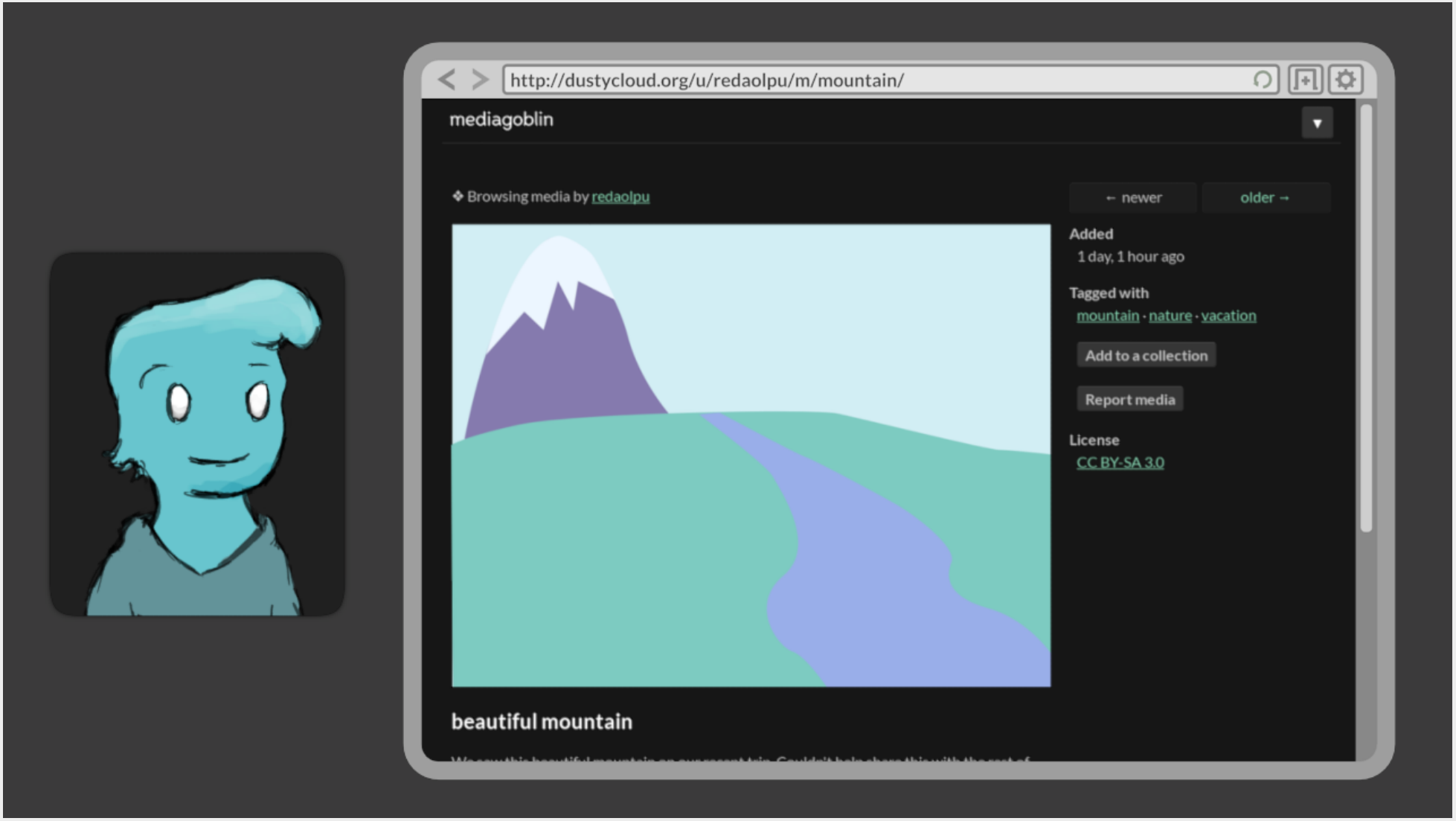


The web we got

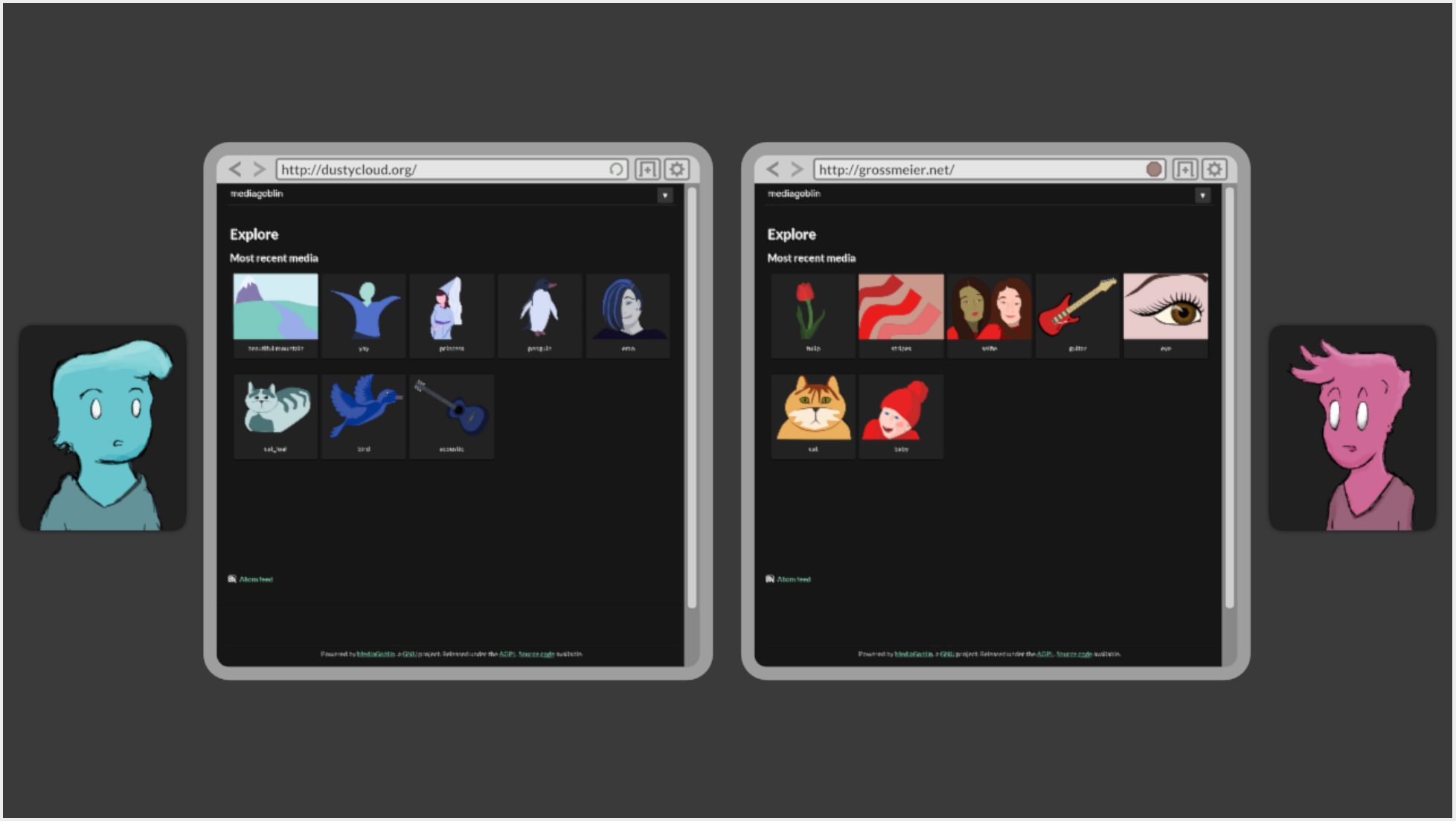


The web we got

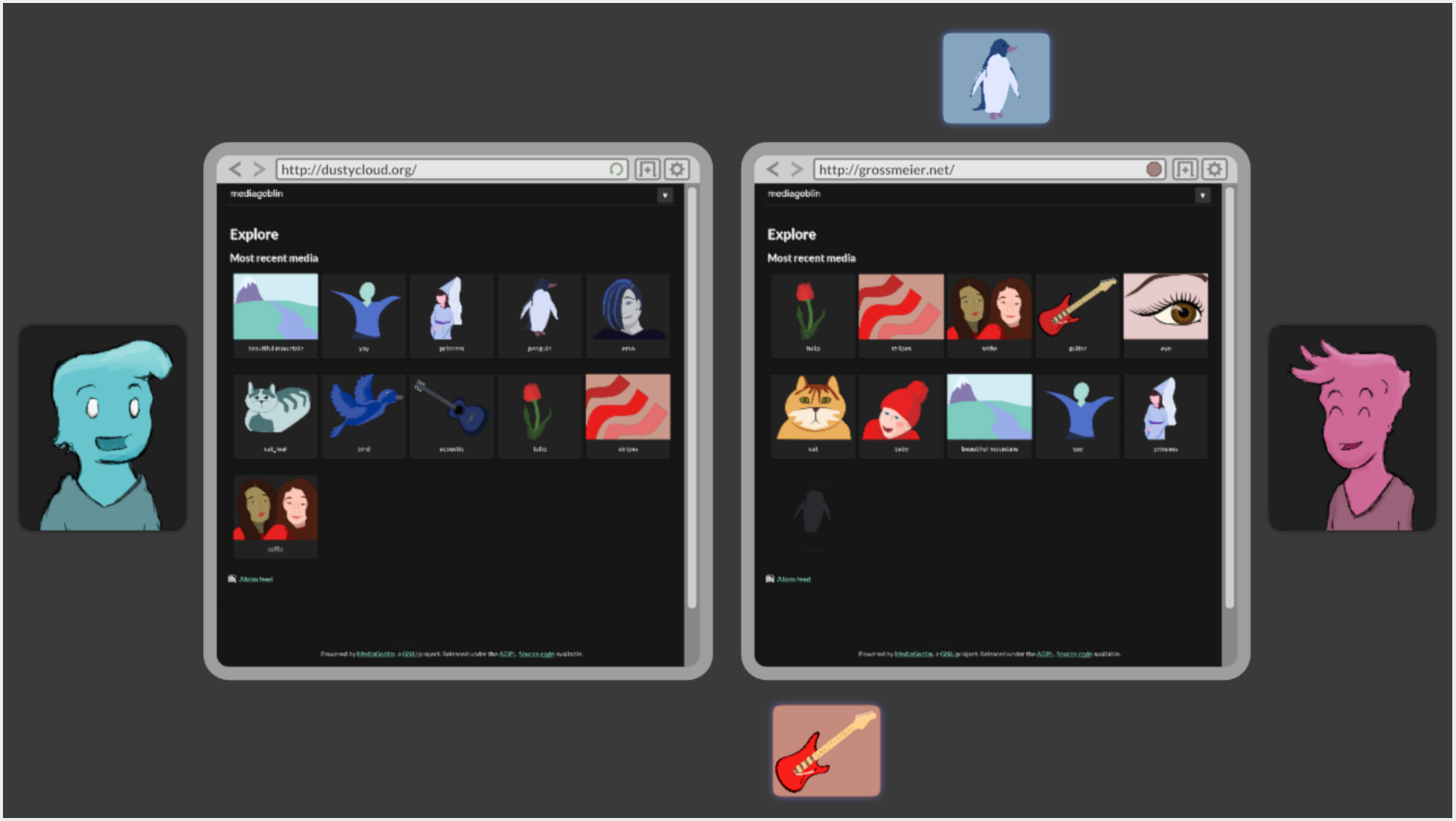
Federation + Self Hosting to the Rescue



Federation + Self Hosting to the Rescue



Federation + Self Hosting to the Rescue



But wait, how to federate?

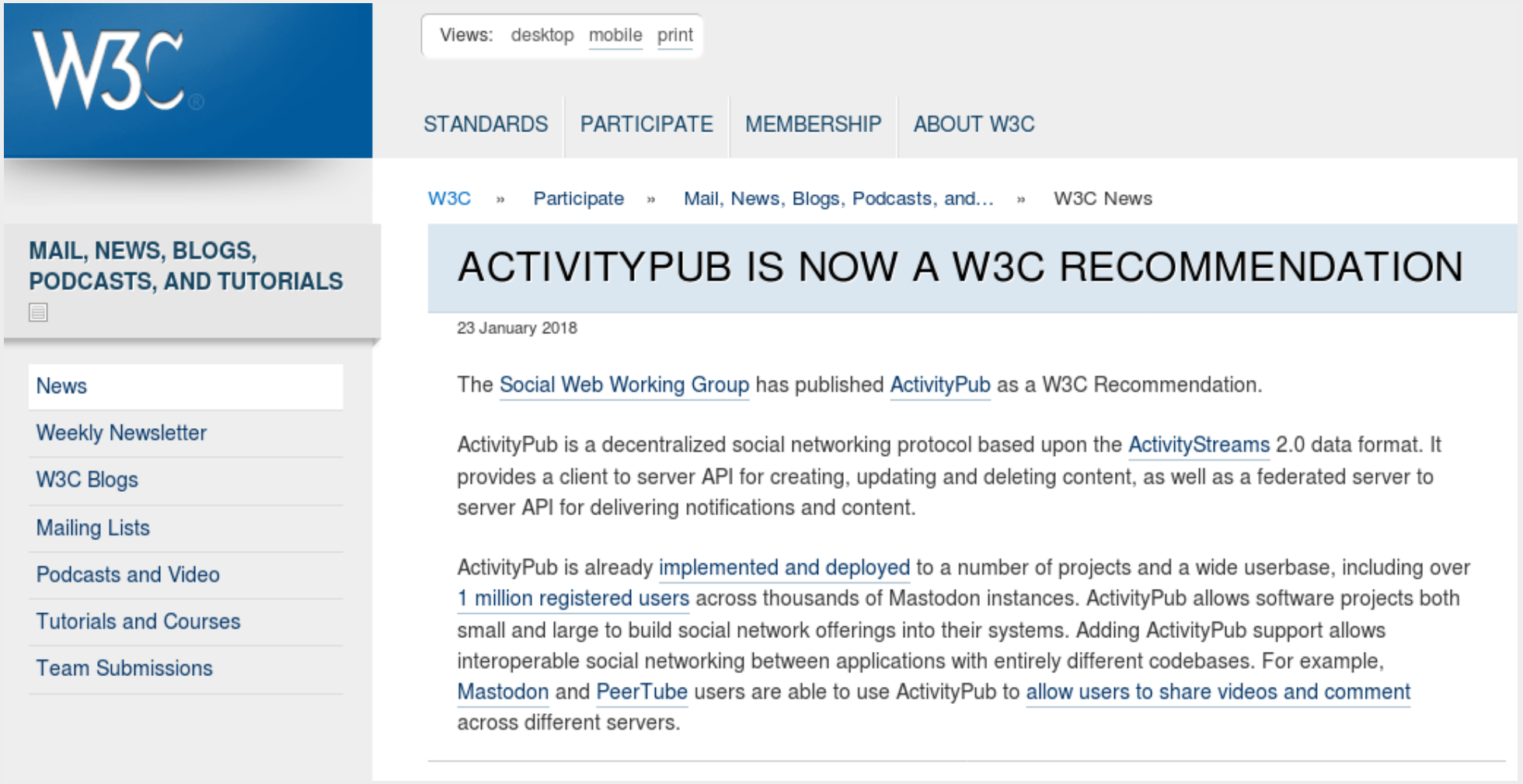
A smattering of incompatible protocols:

- OStatus
- Zot
- XMPP
- Pump
- Tent

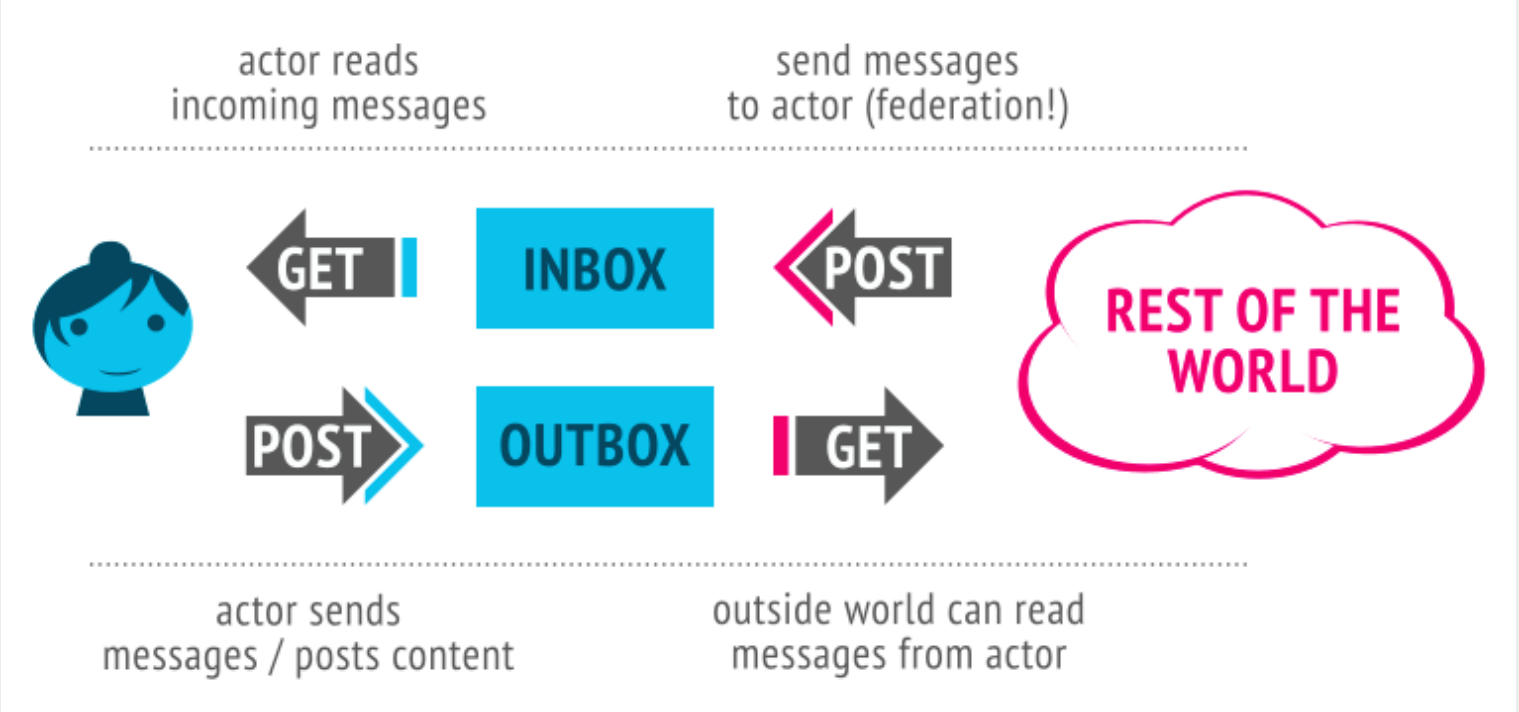
What to do?

ActivityPub

A federation protocol for the web!



Officially standardized, too!



Provides a simple JSON based:

- Server-to-Server protocol (federation!)
- Client-to-Server protocol (mobile, desktop, web clients)

The last year has been huge...

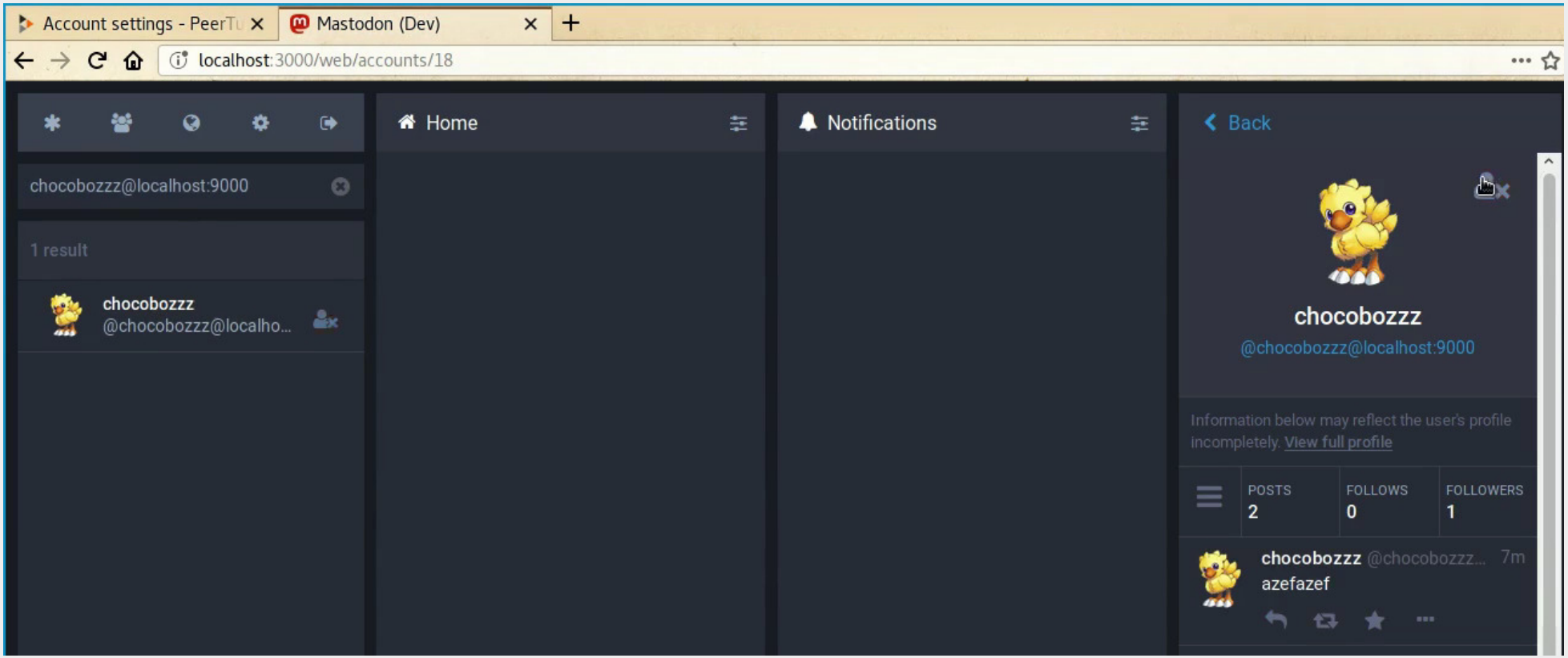
Over 1.5 million registered users

Dozens of implementations:

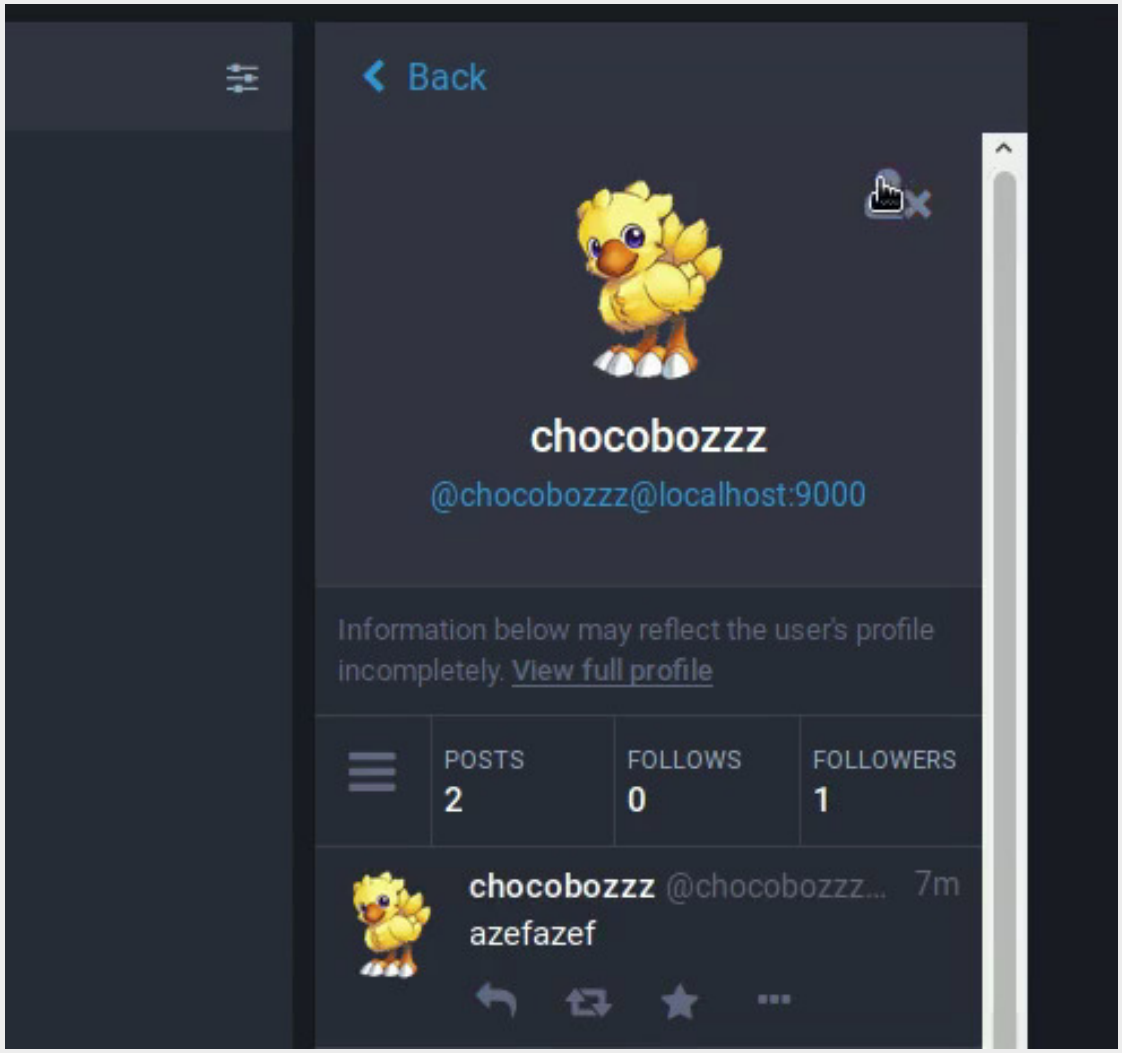
Aardwolf	Kitsune	Pubstrate
Artodon	Koype	Rustodon
Bridgy Fed	Kroeg	Smilodon
CommonsPub	Mastodon	tags.pub
distbin.com	Misskey	Pleroma
Dokieli	Nextcloud	Plume
Funkwhale	Pixelfed	Prismo
Friendica	PeerTube	Pterotype
Hubzilla	places.pub	Wordpress (plugin)
	Pleroma	

... and ActivityPub has only been standardized for 1 year!

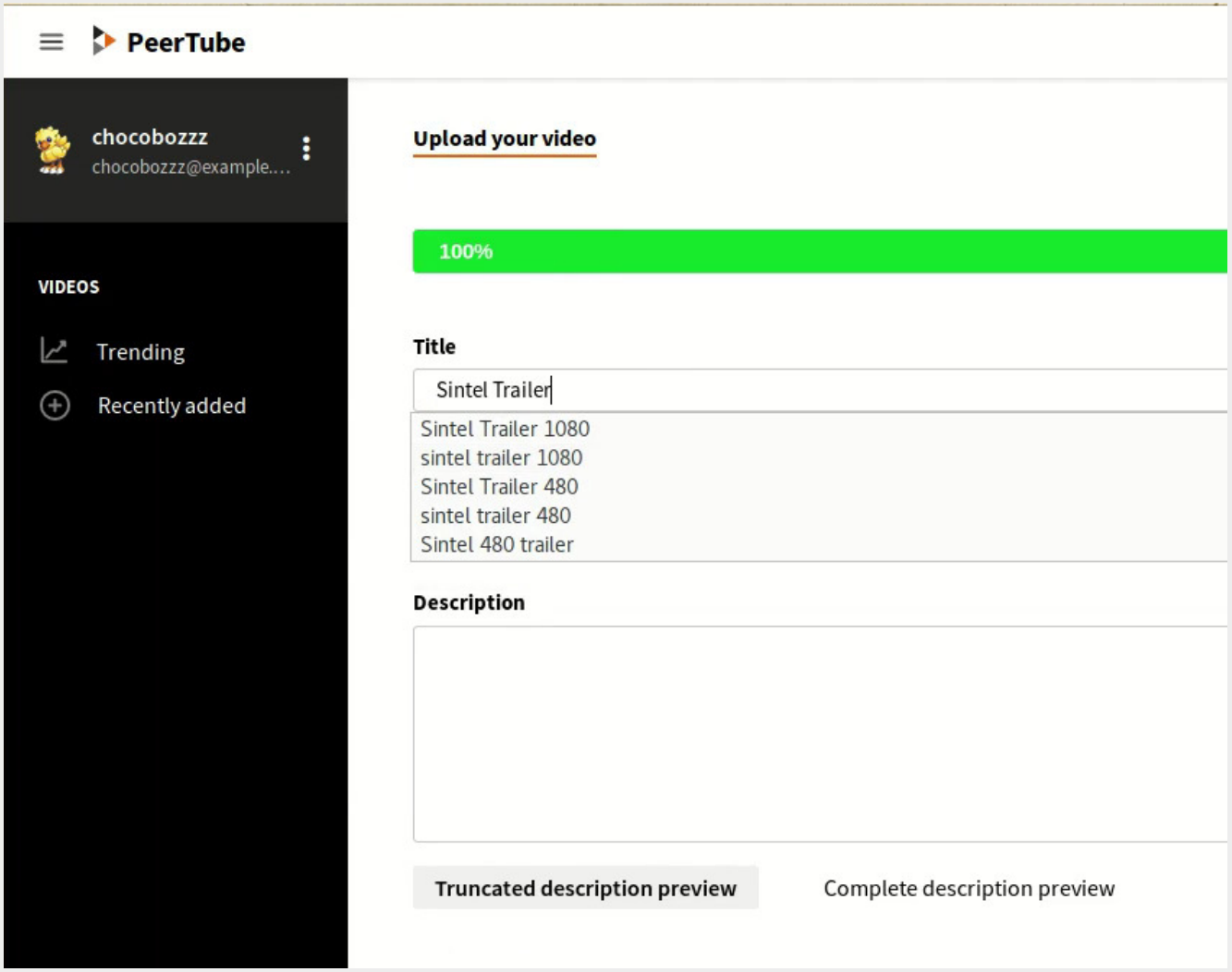
Peertube and Mastodon interop demo!



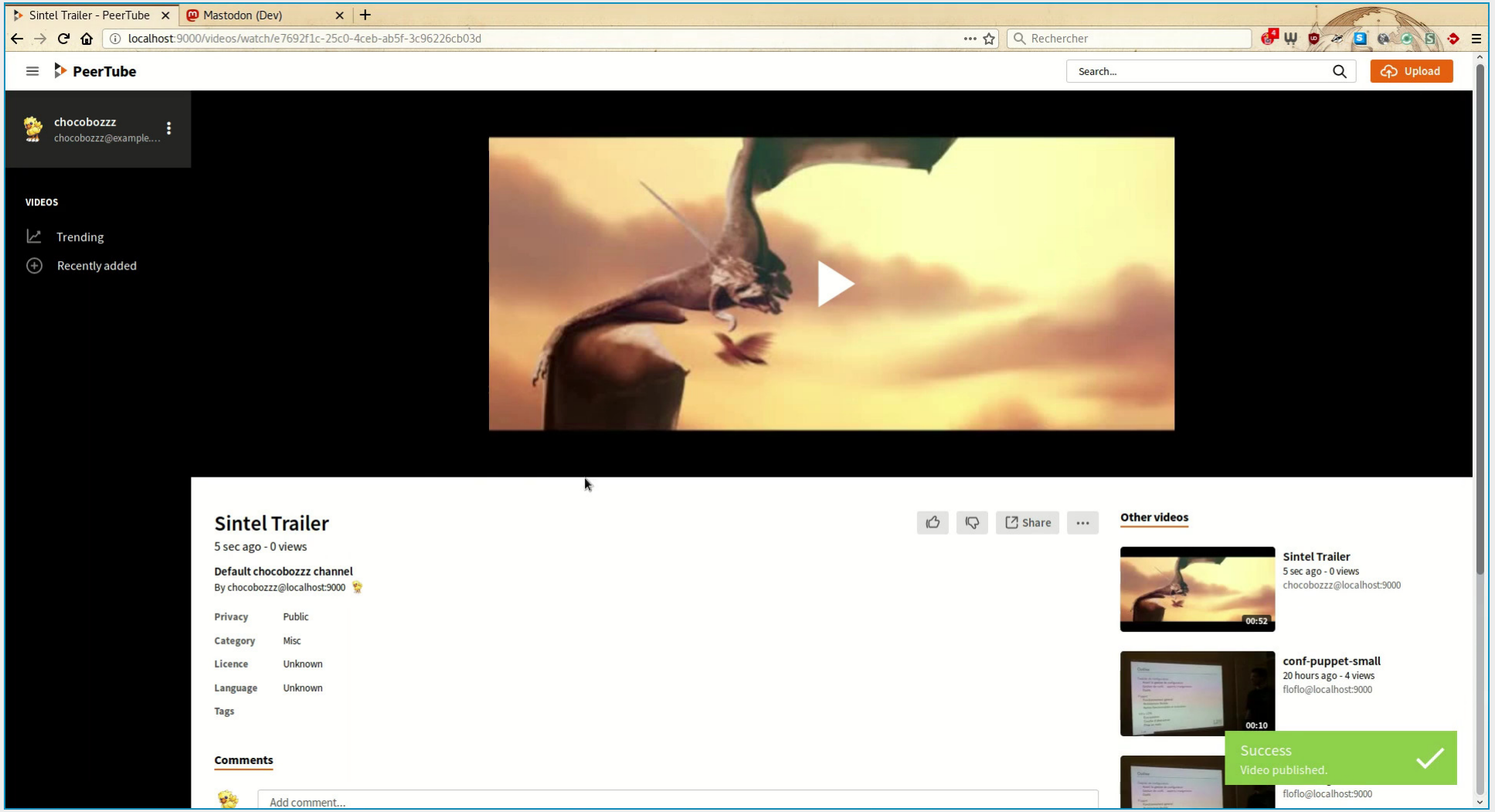
Peertube and Mastodon interop demo!



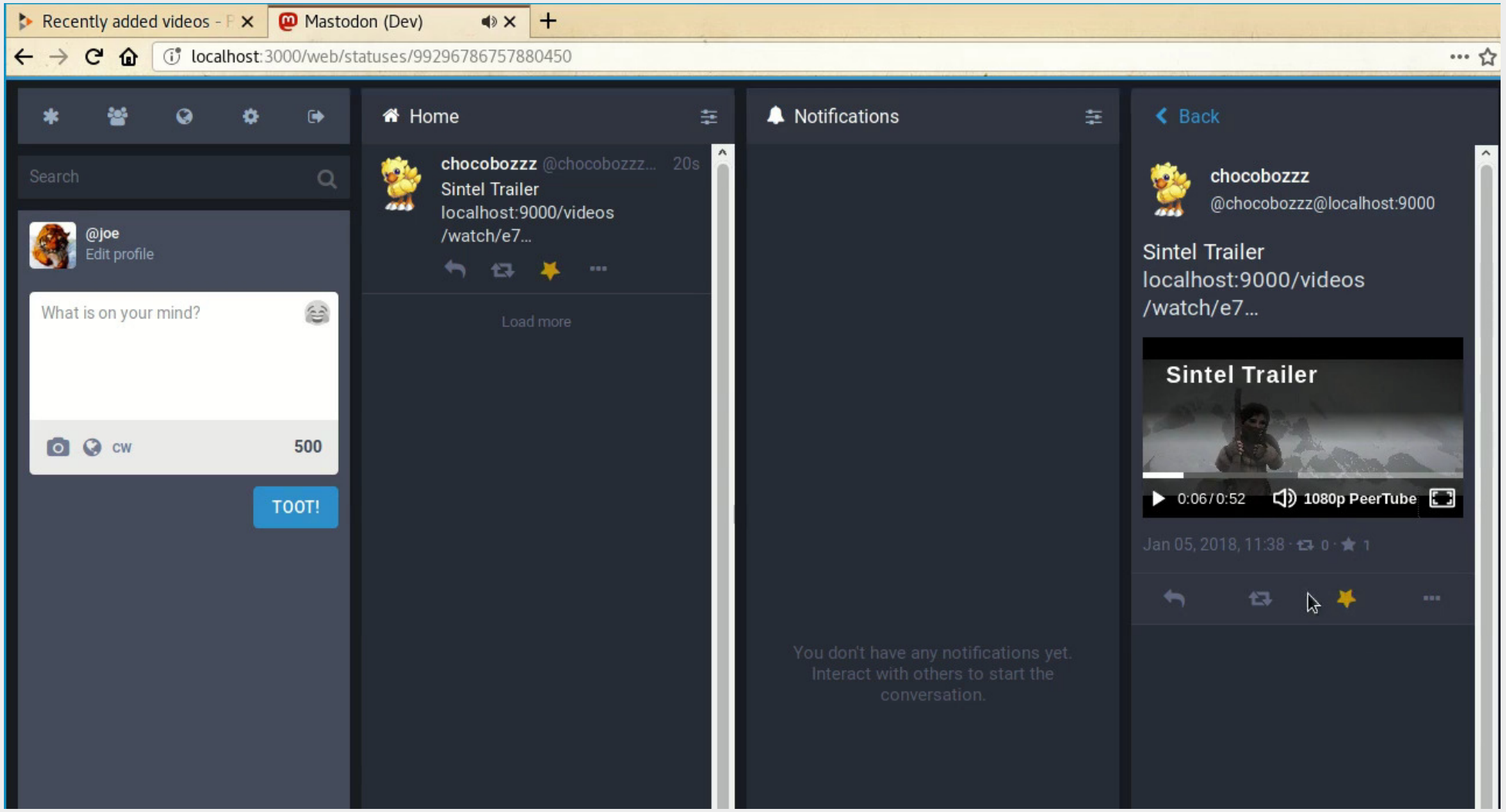
Peertube and Mastodon interop demo!



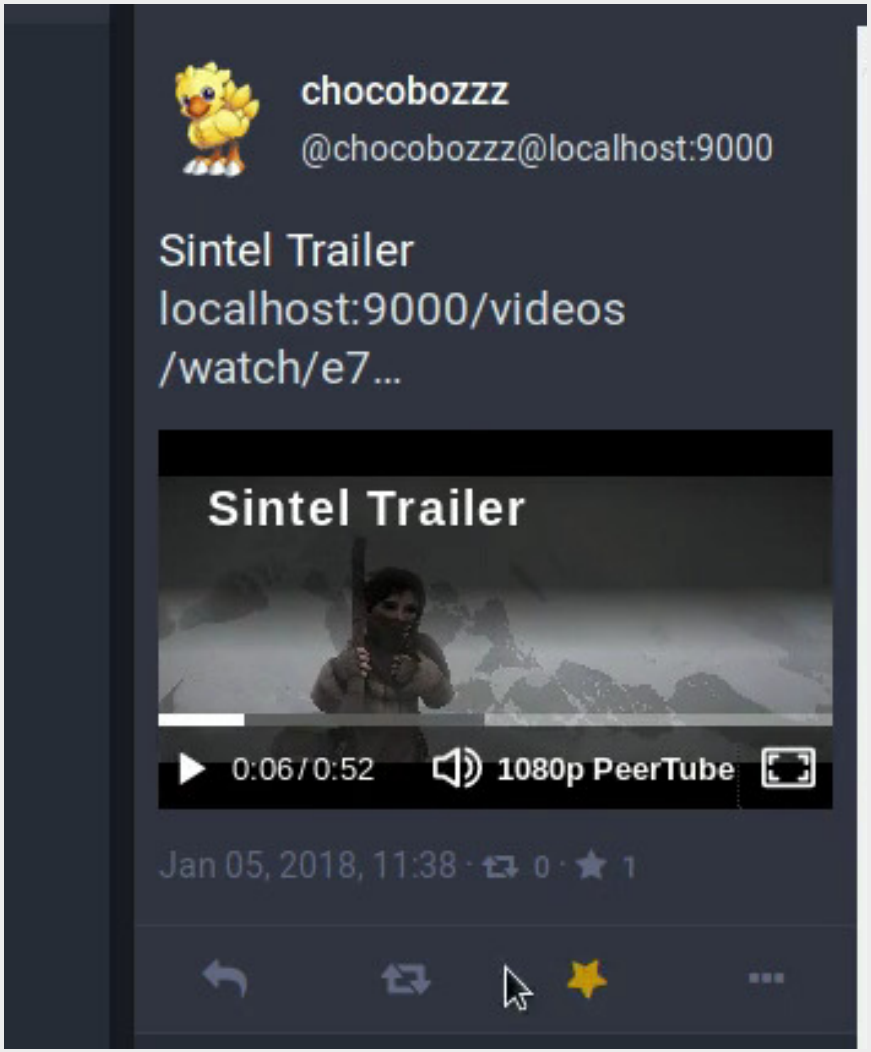
Peertube and Mastodon interop demo!



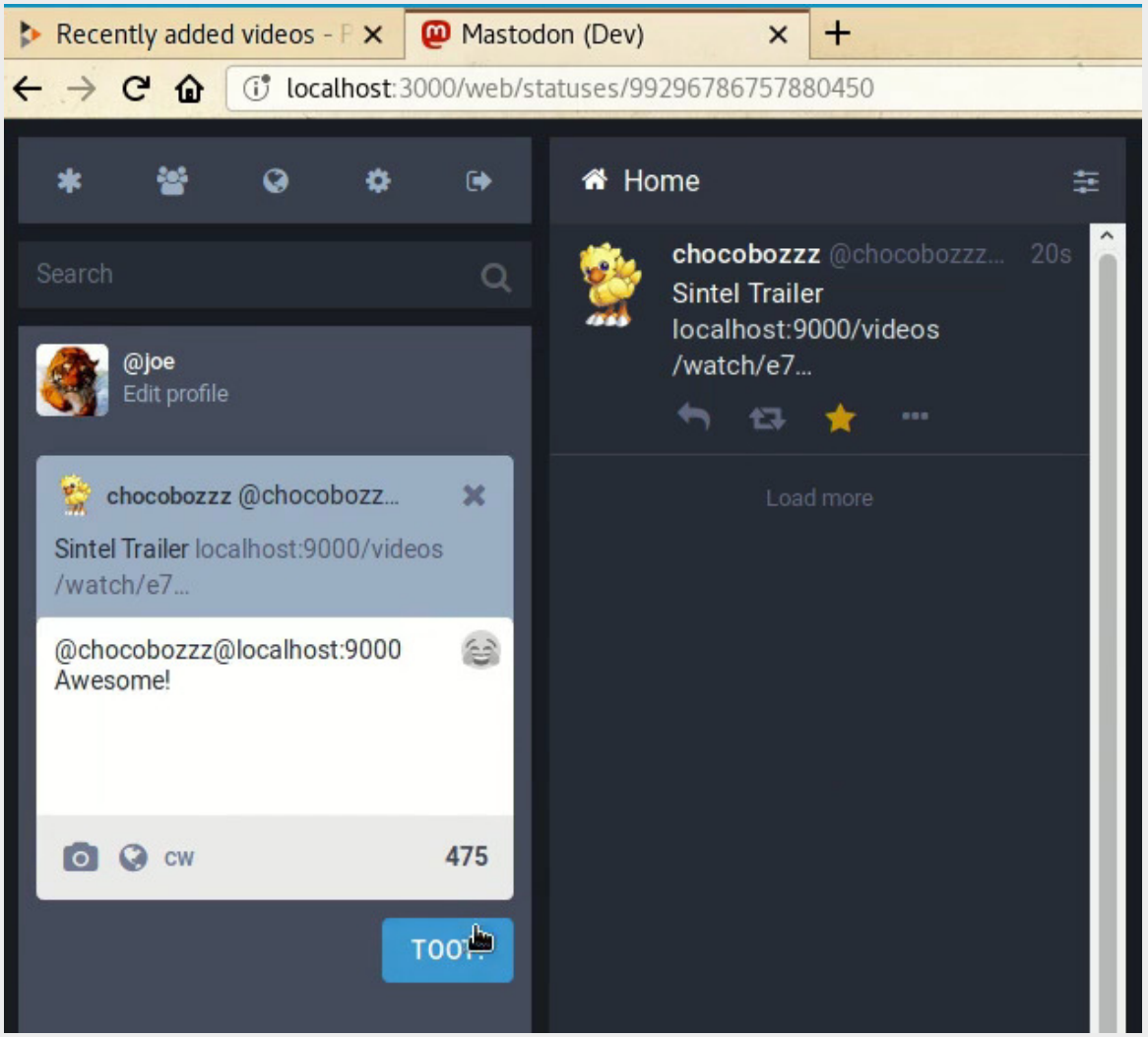
Peertube and Mastodon interop demo!



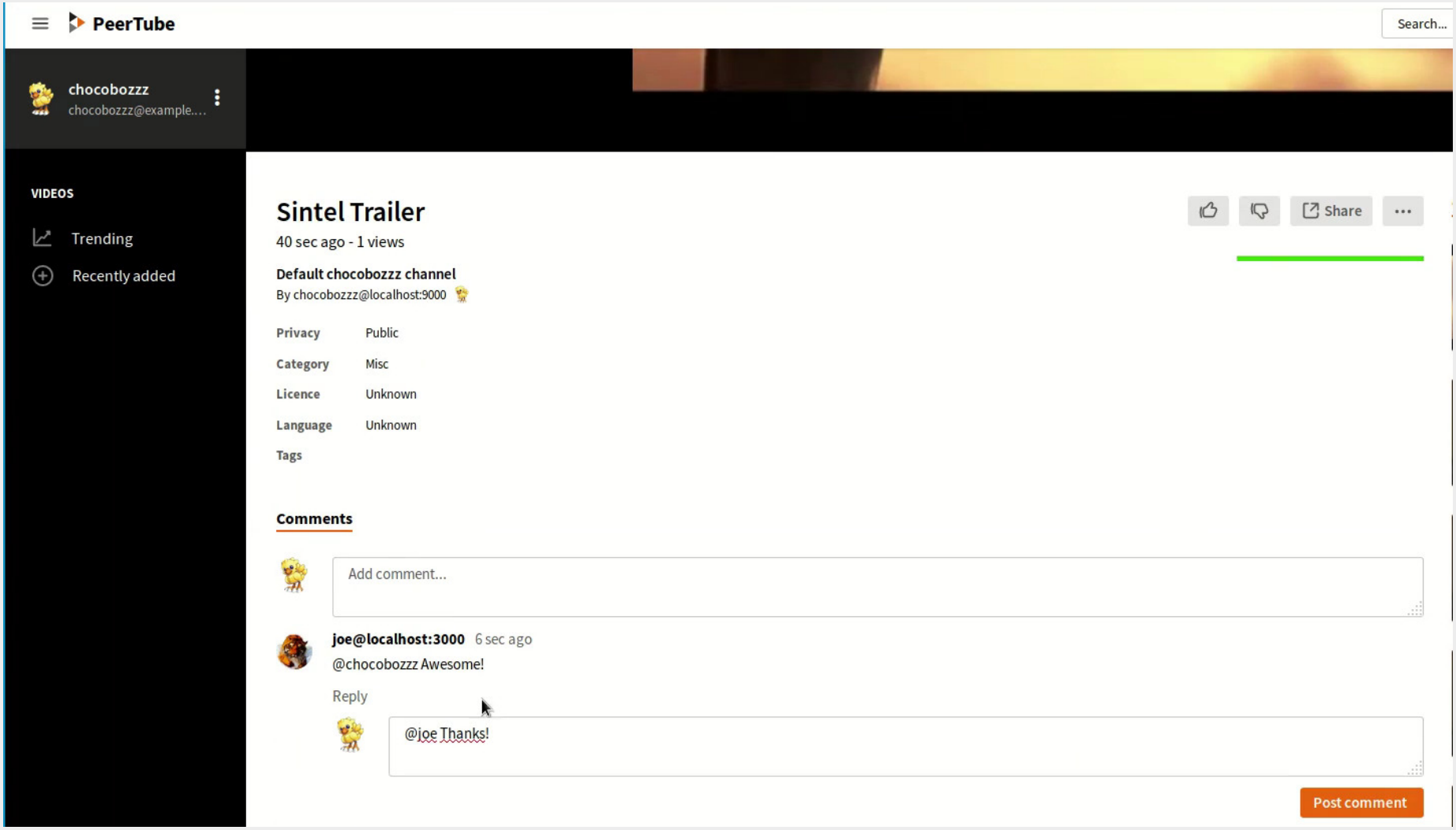
Peertube and Mastodon interop demo!



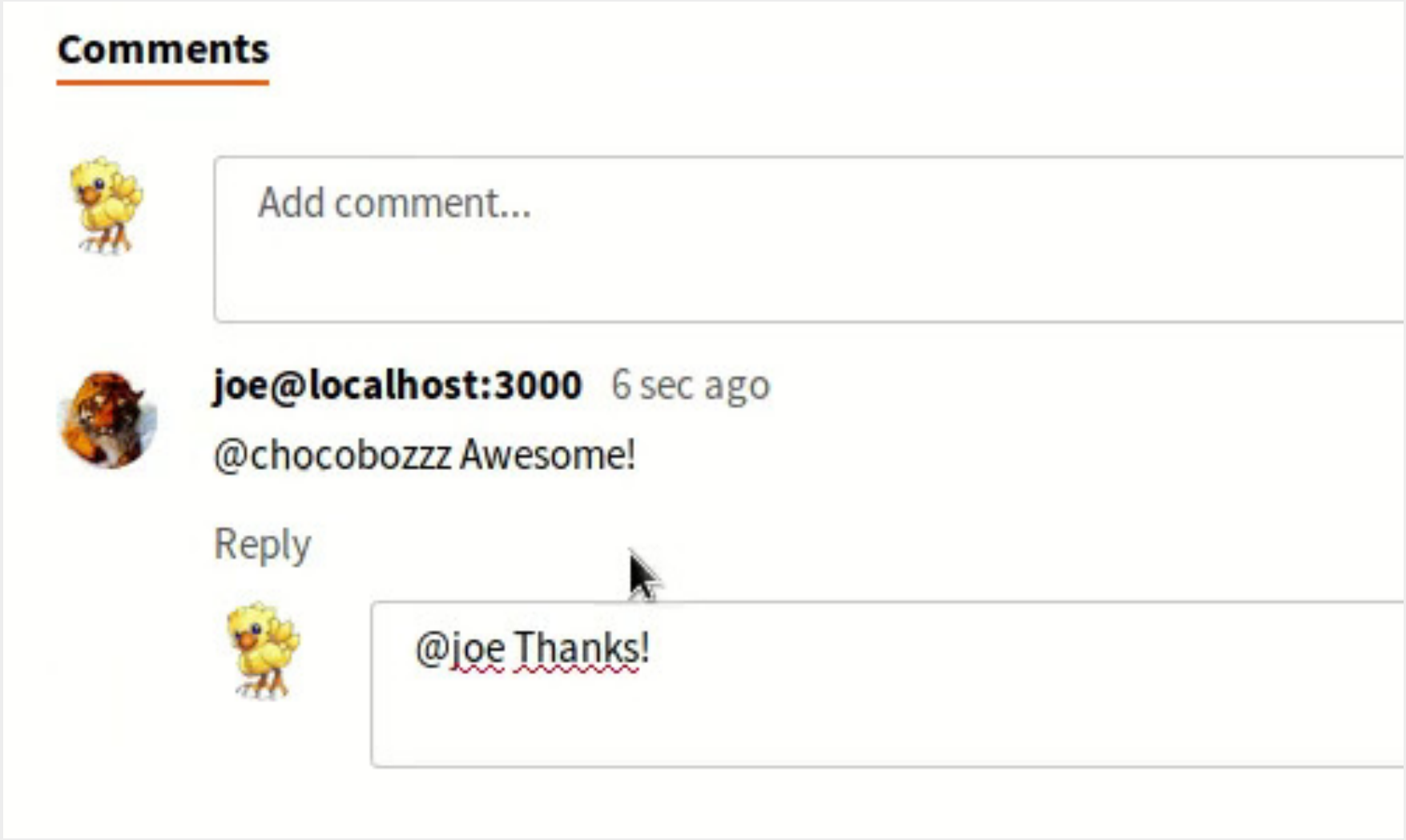
Peertube and Mastodon interop demo!



Peertube and Mastodon interop demo!



Peertube and Mastodon interop demo!



Problem solved??!!!

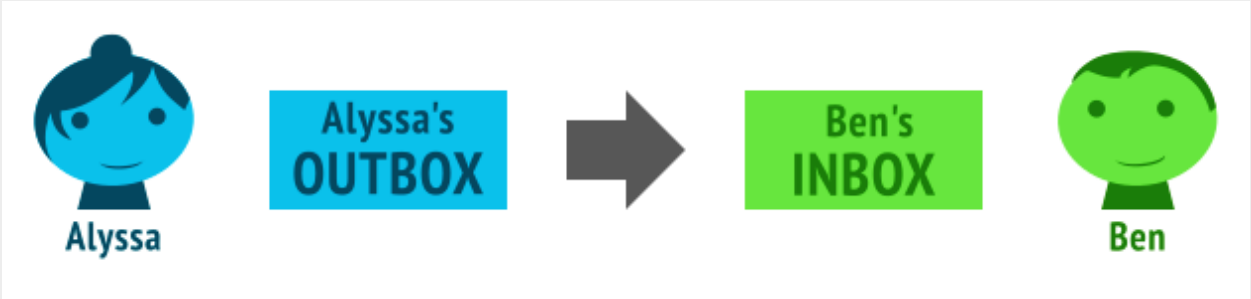
What current implementations can’t do

- Content that survives a server going down
- Highly secure, rich interactions
  - Private photo gallery...
  - ... where you give access to add but not delete items
  - Virtual worlds / games

But these are possible!

ActivityPub did something right...

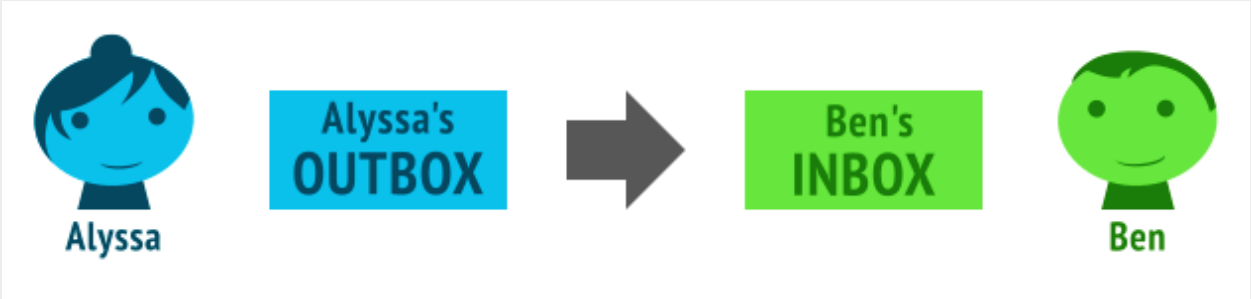
ActivityPub is an actor model protocol



AKA message-passing-centric protocol

It (mostly) doesn’t matter where an actor lives!

ActivityPub is an actor model protocol

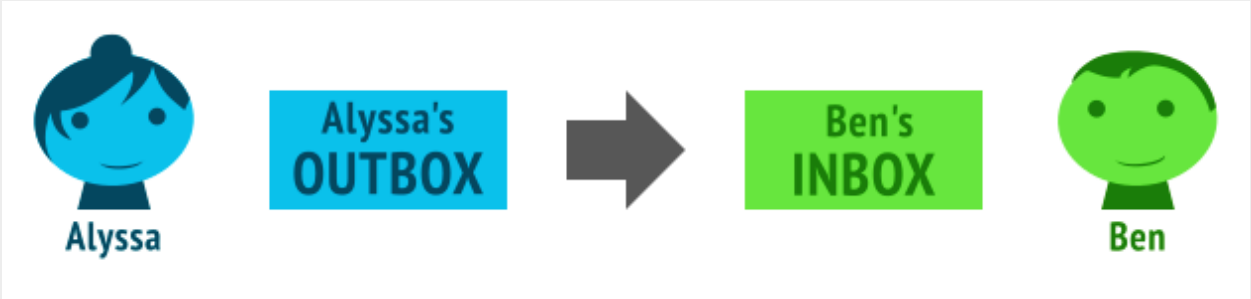


Formal definition...

In response to messages, an actor can:

- Send messages
- Create new actors
- Change its own behavior/state

ActivityPub is an actor model protocol



Many protocols are actor model protocols...

but not all are aware of this.

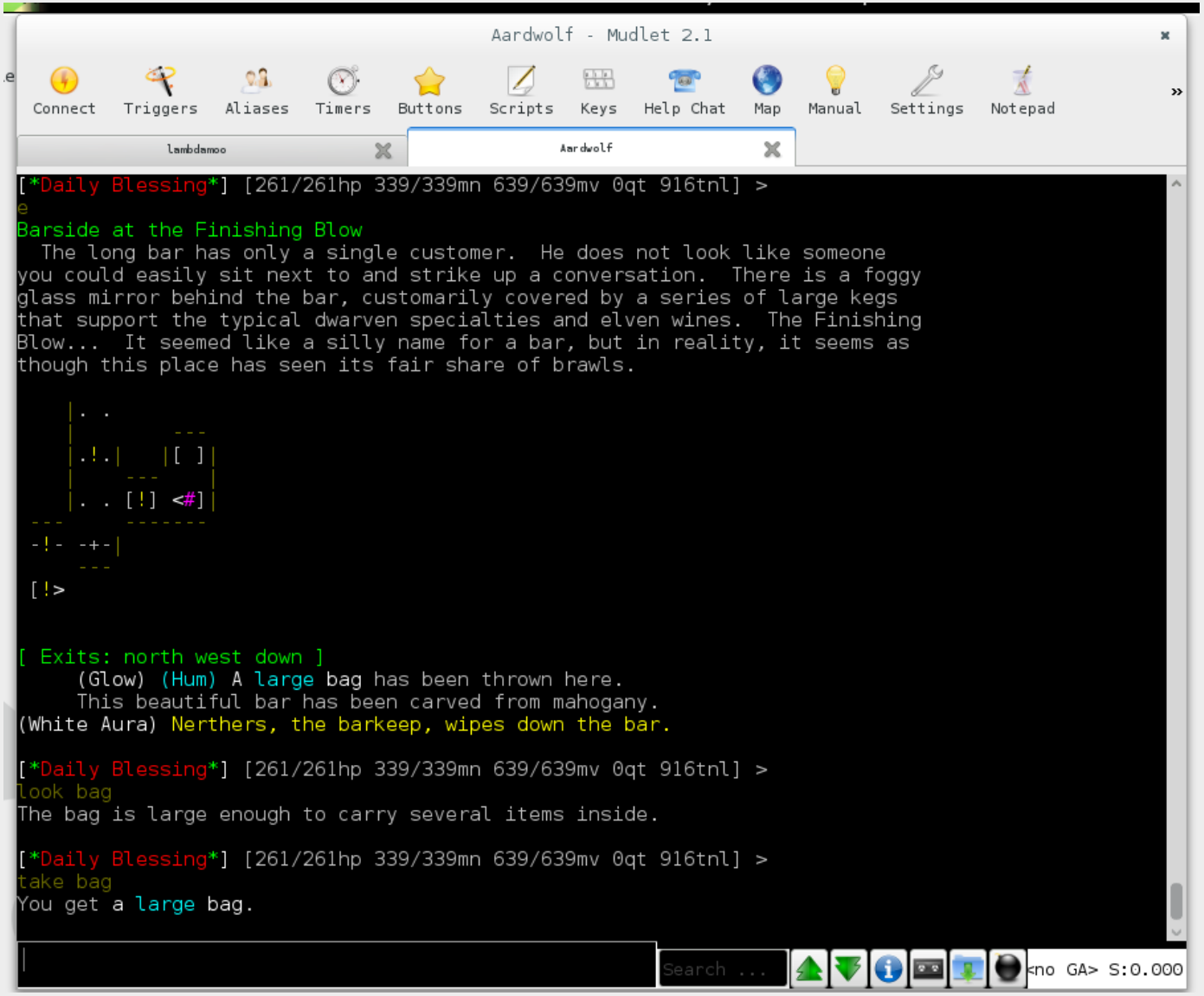
Being self-aware helps!

Spritely

- A federation skunkworks in the public interest!
- Taking the actor model seriously!
- Adding object capability security!
- Stress testing the thing by building distributed games (??!?!)
- Avoiding vaporware: releasing regular artifacts/demos

The lost cyberpunk world of social games

MUDs, MOOs, MUSHes: social spaces with a sense of place



The lost cyberpunk world of social games

Habitat: massively multiplayer graphical game... in 1985!

https://web.stanford.edu/class/history34q/readings/Virtual_Worlds/LucasfilmHabitat.html

The lost cyberpunk world of social games

Menu

mailto:vijay@saraswat.o

Converted to HTML: Sun Oct 5 1997
Last modified: Wed Apr 23 1997
Started work: Mon Apr 21 1997

Design requirements for network spaces

Vijay Saraswat
AT&T Research , 180 Park Avenue, Florham Park NJ 07932

April 1997

Introduction

A wide variety of network communities exist today, supported by many different computational platforms. As the need for new architectures for these platforms arises, so does the need to articulate what exactly should these platforms provide, abstracted from the many (and somewhat diverse and divergent) concrete realizations of these ideas (e.g. in MOO, MUSH, MUD etc).

This note presents my analysis of the desiderata for ***network spaces*** , my generic term for the computational platforms underlying network communities. My analysis is not driven by any attempt to understand a "least common denominator" for these different approaches. Rather it is driven by my experience starting, administering and participating in several such communities since 1994 and from my desire to find a coherent and consistent conceptual framework (e.g. one that resolves issues of objects, persistence, identity, change etc) within which system development may proceed interlinked with, and yet somewhat decoupled from, the diversity of network communities that may arise atop such spaces.

I believe this task (of articulating the desiderata of network communities) is of some urgency. Conditions are now ripe for an appropriately designed architecture and implementation to provide the basis for the development of tens of thousands of interlinked network communities all over the globe. On the side of social sciences research the extraordinary interest of these spaces as both a synthetic and analytic tool for the study of communities is now rapidly becoming evident. On the computational side the development of MOO as a basis for such spaces has come to a halt with the disintegration of groups working on this technology at PARC. On the other hand, the rapid maturation of Java and CORBA technology, and widespread deplyoment of networked personal computers is finally(!) providing the ubiquitous basis on which large-scale end-user populated distributed systems may be realized. Therefore this task is both timely and important.

1. In a nutshell, a network space provides an object-oriented, networked, multi-person, persistent, programmable, multi-interface, distributed infrastructure for the construction of administered, shared, extensible virtual worlds. (The LambdaMOO server, running, say, with JHCore, is an example of such a system, and should be kept in mind in the following discussion.) In this, network

This existed!

Electric Communities Habitat

Secure, decentralized virutal worlds!

Sadly little survived, except...

E!

Open Source
Distributed Capabilities

Welcome to *ERights.org*, home of ***E***,
the secure distributed persistent language
for capability-based smart contracting.

Quick Start | What's New? | What's ***E***?
Smart Contracts | History & Talks | Feedback

[California Home] [Mirror in Virtual Tonga]

We do not influence the course of events by persuading people that we are right when we make what they regard as radical proposals. Rather, we exert influence by keeping options available when something has to be done at a time of crisis.

--Milton Friedman

The greatest programming language you've never heard of! Featuring:

- Actors for run-anywhere evaluation
- Local immediate evaluation
- Most importantly: object capabilities (ocaps)!

Contrasting approaches: ACLs Don’t

A dangerous program on your computer...

- It can run any program it wants as you
- It can read all your data
- It can post your secrets to any server

What program is it?

Solitaire! (Or actually any program)

Contrasting approaches: ACLs Don’t

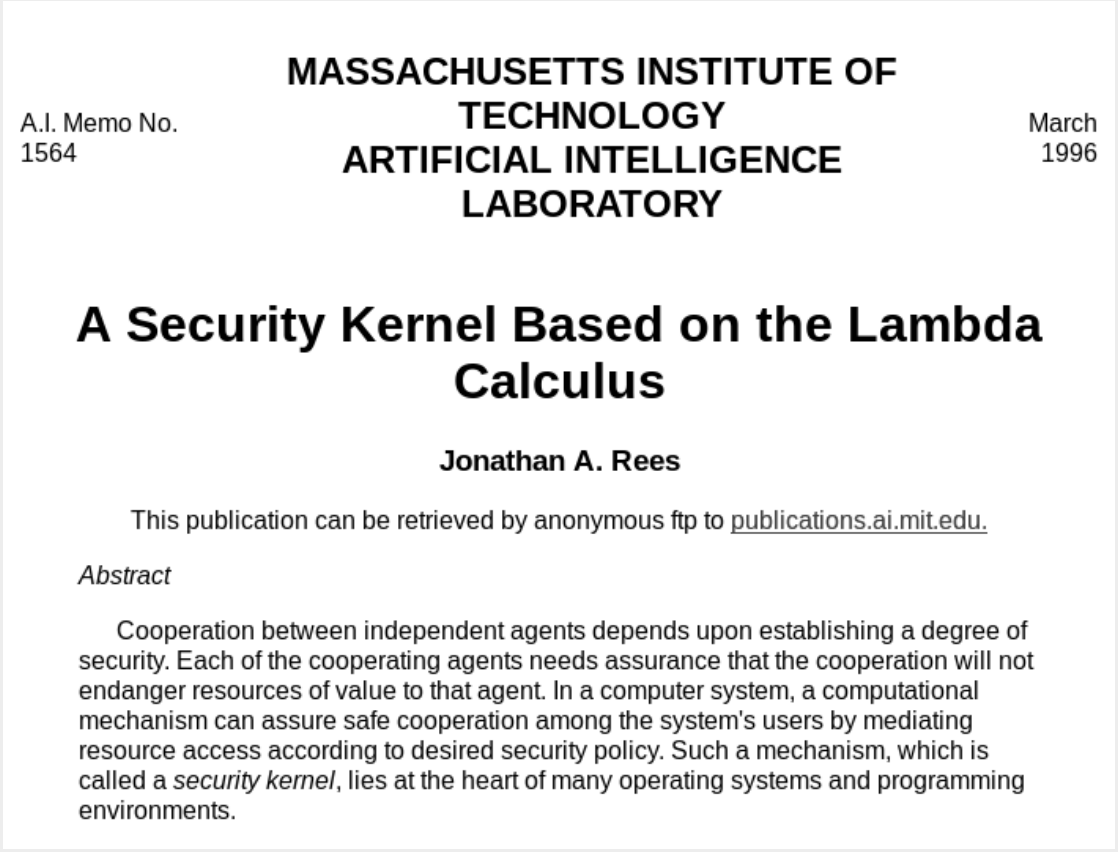
Identity-centric authority doesn’t work

- Ambient/excess authority
- Confused deputy problems (ask me about Guile’s REPL story if we have time)

See the paper "ACLs Don't" for more information

Object Capbilities (ocaps)

How to allow security AND rich interactions



A.I. Memo No. 1564

MASSACHUSETTS INSTITUTE OF TECHNOLOGY ARTIFICIAL INTELLIGENCE LABORATORY

March 1996

A Security Kernel Based on the Lambda Calculus

Jonathan A. Rees

This publication can be retrieved by anonymous ftp to publications.ai.mit.edu.

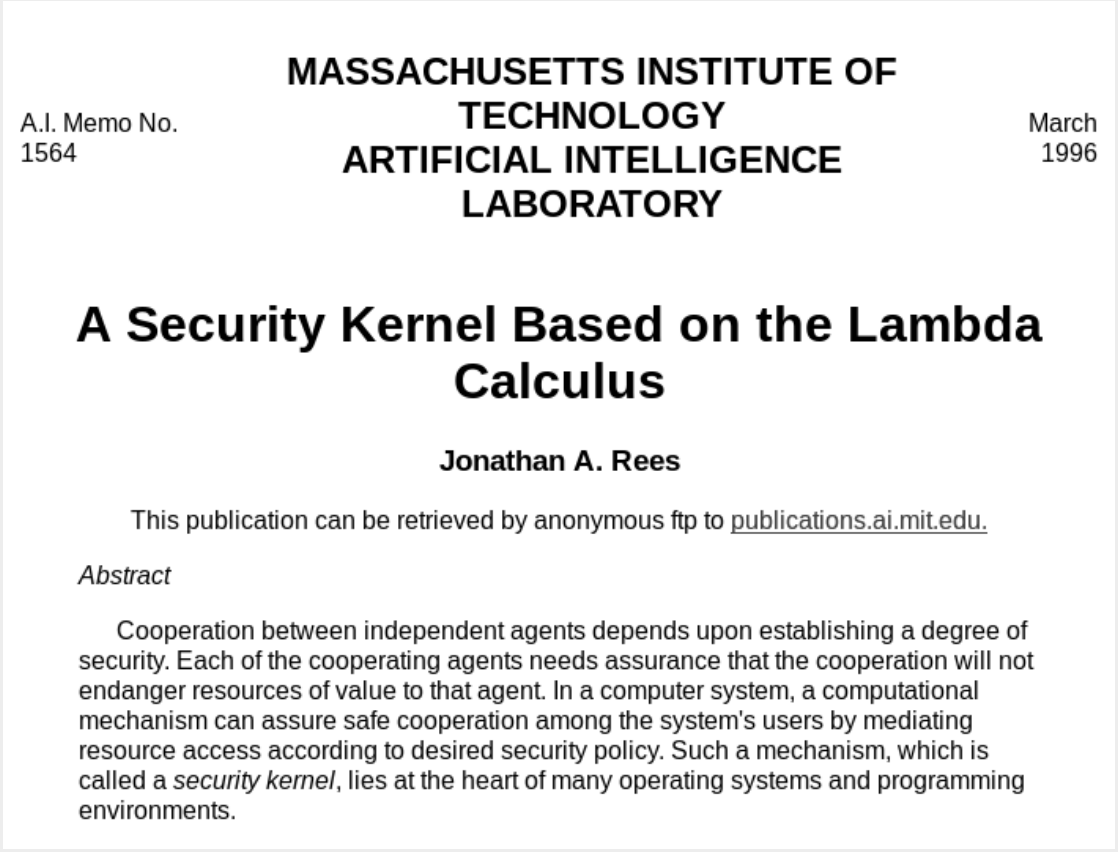
Abstract

Cooperation between independent agents depends upon establishing a degree of security. Each of the cooperating agents needs assurance that the cooperation will not endanger resources of value to that agent. In a computer system, a computational mechanism can assure safe cooperation among the system's users by mediating resource access according to desired security policy. Such a mechanism, which is called a *security kernel*, lies at the heart of many operating systems and programming environments.

- AKA "Lambda, the Ultimate Security Mechanism"
- Lexical scope *is* your security model

Object Capbilities (ocaps)

How to allow security AND rich interactions



A.I. Memo No. 1564

MASSACHUSETTS INSTITUTE OF TECHNOLOGY ARTIFICIAL INTELLIGENCE LABORATORY

March 1996

A Security Kernel Based on the Lambda Calculus

Jonathan A. Rees

This publication can be retrieved by anonymous ftp to publications.ai.mit.edu.

Abstract

Cooperation between independent agents depends upon establishing a degree of security. Each of the cooperating agents needs assurance that the cooperation will not endanger resources of value to that agent. In a computer system, a computational mechanism can assure safe cooperation among the system's users by mediating resource access according to desired security policy. Such a mechanism, which is called a *security kernel*, lies at the heart of many operating systems and programming environments.

- You can only access what you hold a reference to
- Works in protocols or languages!

Another metaphor

Ye olde ocap car key metaphor goes here

(Partially) written Spritely artifacts/demos

- Goblins: actor model library for Racket
- Magenc: private and encrypted p2p shareable content
- Petnames paper for RWoT written, demo to come
- Pre-ocap MUD demo

Goblins: an actor model library for Racket

- Spawn actors, send messages to actors
- Currently local-only, very soon multi-machine
- Why a library? Why not a #lang?

Enough smalltalk! Let’s see a demo!

I see what you did there

- Spawning a lambda actor
- Spawning a class’y actor
- Sending a message: basic
- Splitchronous send with <<-
- Promises!

<<-, friend or foe?

- Very convenient!
- Uses delimited continuations under the hood
- But... re-entrancy attacks?
- I’m conflicted, help me out language folks

But is it ocap-secure?

Racket's lexical scope: safe

Racket's module importing: not safe

Future artifact: Dungeon

Thanks! Questions?

- Personal site: https://dustycloud.org/
- Fediverse: https://octodon.social/@cwebber
- Birdsite: https://twitter.com/dustyweb/
- Support this work: https://patreon.com/cwebber