Pump.io + Community = ♥

AJ Jordan

Let's talk about moi

Arch Linux

Emacs

Git

zsh

Node.js

Firefox

Also about moi

Radical FLOSS zealot

Self-hoster

Pump.io primary maintainer

Gap year (mostly) @ the Recurse Center

he/him

Let's talk about this session

What & why of pump.io

Its API

New community (+ history)

The "what & why"

Social network

Federated

Successor to StatusNet (aka GNU Social)

"That thing that runs identi.ca"

Pump.io != StatusNet++

Both written by Evan Prodromou

StatusNet was his first social project

identi.ca circa July-ish 2013

Very complex

Codebase problem & protocol problem

(Side note: Apache 2.0 vs. AGPL)

StatusNet protocol

(Fair warning: this is Long Before My Time™)

OStatus

Atom (status update feed)

PubSubHubbub (post push)

Salmon (comment distribution)

ActivityStreams (favorites, etc.)

Let's talk Salmon

Just as an example.

Comment distribution

Based on Atom/RSS

Data is POSTed back to an endpoint URL in original post

Turns out this is overkill

StatusNet code problems

Everything and the kitchen sink

OpenID

Twitter cross-posting

URL shortening

Publish updates over XMPP

Content embedding (think YouTube)

A lot of this had custom server-side stuff

140 characters

Looked like Twitter

Artificial limits

Eventually evolved to support pictures, etc.

AKA: understood more and more types of ActivityStreams

Enter pump.io

I post something and my followers see it. That's the rough idea behind the pump. - README.md

In other words:

Pump.io is just an ActivityStreams distribution engine

ActivityStreams

Usually JSON

Actor, verb and object (+ optional target)

```
"published": "2011-02-10T15:04:55Z",
"actor": {
    "url": "http://example.org/martin",
    "objectType": "person",
    "id": "tag:example.org,2011:martin",
    "image": {
        "url": "http://example.org/martin/image",
        "width": 250,
        "height": 250
    },
    "displayName": "Martin Smith"
```

The API

OAuth 1.0

Post activities to the user's outbox

Read activities from the user's inbox

Major/minor stream differentiation

Example: let's post a note

POST to my outbox

Pump.io POSTs to my follower's inboxes

Generic ActivityStreams distribution

OpenFarmGame

Implements custom verbs and objects

Pump.io distributes them, no problem

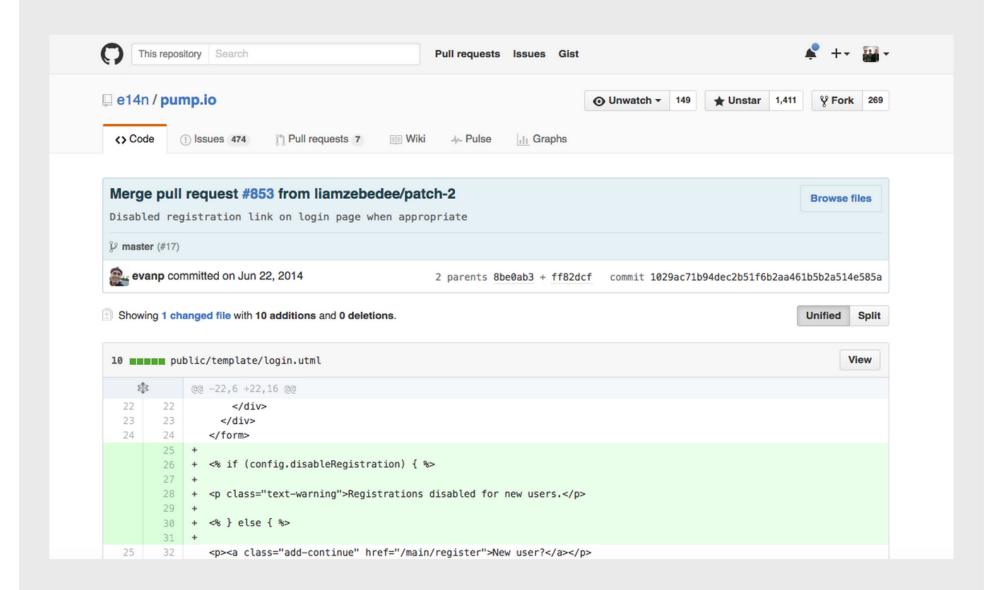
Generic, extensible social server

I post something and my followers see it. That's the rough idea behind the pump. - README.md

Community

June 22, 2014

Evan pushes the last commit for over a year

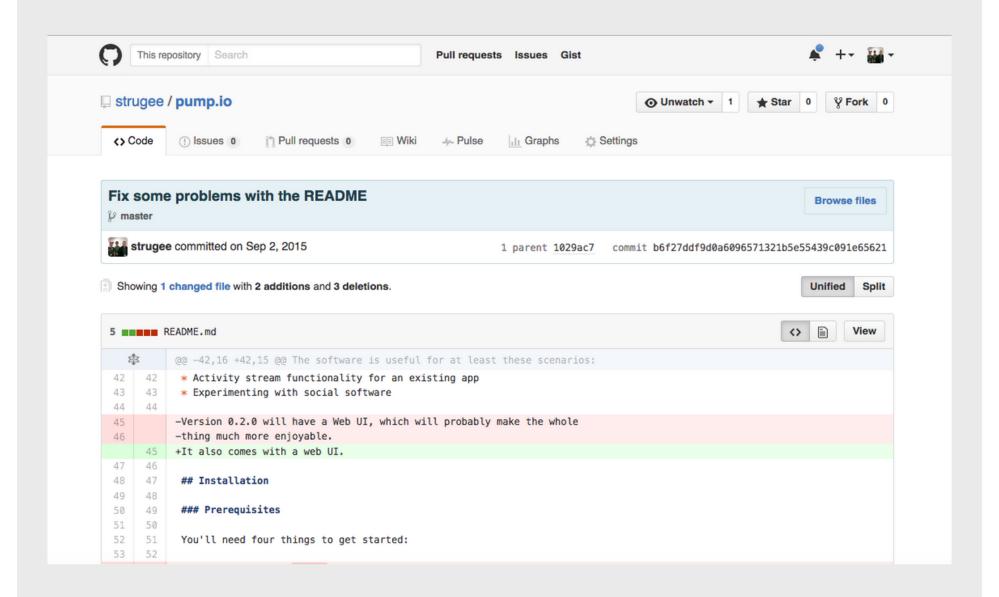


The fork

Late August: I send this note to Evan, asking him to nominate a new maintainer

I fork <2 weeks later

First commit is September 2, 2016



Bringing it home

September 21, 2015: Evan wants to hand off the code to the community

Informal meeting that same day

Evan grants admin rights to cwebber and jpope

Since then...

I become primary maintainer

Small but strong community

Meetings every third Friday of the month

PRs get merged now

Releases

Focus so far: reducing technical debt

Almost 0 outdated dependencies, perf improvements, bug fixes

Releases every month, alternating betas/stables

4.0 entering beta April 1

Other awesome stuff

Conservancy

Node transition

Community accounts

MediaGoblin federates with us!

SocialWG (ActivityPub in PR, ActivityStreams 2.0 in PR)

Future

ActivityPub support

Out-of-the-box HTTPS

More security improvements

Packaging

Bugfixes

Join us!

#pump.io on Freenode

XMPP: pump.io@muc.jappix.com

pump-io/pump.io on GitHub

@pumpdotio or (even better) pumpio@pumpit.info

Sign up for a hosted node...

...or run your own!

Thanks!

I've been AJ

alex@strugee.net (email & XMPP)

alex@pump.strugee.net

@strugee2 (if you must)

strugee on GitHub

strugee.net

Access this presentation again

https://strugee.net/presentation-pumpio/libreplanet/



Or, check out the source.