Towards peer-to-peer alternatives for code collaboration

@cloudhead

Monadic
Let’s talk *git*.

The common ground for code collaboration.
What's missing?

Discovery
Where to find changesets pertaining to a project?
Where to find peers replicating a repository?

Canonicity
Where should I pull from?
Who has the latest copy?
Where is it hosted?

Social artefacts
Issue tracking and code review aren't provided by git.
<table>
<thead>
<tr>
<th>Where to see all patches?</th>
<th>Where to pull from?</th>
<th>How to discuss changes?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mailing list.</td>
<td>Linus.</td>
<td>Mailing list!</td>
</tr>
</tbody>
</table>

What about user experience?
Hosted platforms

solution #2

Where to see all patches?

Where to pull from?

How to discuss changes?

<3 Bill
Does peer-to-peer have an answer to code collaboration?
Why peer-to-peer?

**Economic resilience**
Peer-to-peer systems don’t need to “make a profit” to subside. The burden is distributed evenly.

**Political resilience**
Peer-to-peer systems aren’t subject to the will of authorities. They cannot be captured.

**Technical resilience**
Peer-to-peer systems are harder to attack. There is no single point of failure.
Let's look at Secure Scuttlebut

a resilient, peer-to-peer social network
What about code collaboration?

git + p2p = ?
Radicle
git + peer-to-peer replication and discovery

[Diagram of a seed with labeled parts: Seed coat, Endosperm, Cotyledons, Elaisome, Radicle, Embryo, Ricinus communis]
**Radicle**

‘tis the root

**Offline-first**

Your issues, comments and reviews live on your machine. Everything is always available, even offline.

**Secure**

Secure peer identities (EdDSA). All artefacts are cryptographically signed, including code review.

**Peer-to-peer**

No hassle setting up your own hosting, and no trusted third-parties.

SHA-1 is broken
Radicle
foundations & inspiration

SSB
Secure Scuttlebut’s “social overlay” protocol. Replication is based on a follow graph.

TUF
The Update Framework. Secure software updates even when keys are compromised by an attacker.

Git
The foundation upon which Radicle is currently built.
Git as a foundation for peer-to-peer

**Offline-first**
Optimized for full replication.
Git doesn’t assume an always online server.

**Designed for the bazaar model**
Which is awfully similar to how peer-to-peer networks are structured.

**Packfiles**
An interactive protocol that makes synchronization fast.

git gossips well
Radicle

architecture (briefly)

Working copy
- master
- remotes/alice
- remotes/bob

Replicated copy
- src/master
- src/remotes/alice
- src/remotes/bob
- rad/remotes/bob
- rad/remotes/bob/remotes/ange
- rad/remotes/bob/remotes/luke
- rad/project
- rad/contributor
Radicle

Radicle

commit bf0c3d41b54fb7b98aada9ee1df5fa6bb6adc08c (rad/project)
Author: Alexis Sellier <alexis@monadic.xyz>
Date:   Tue Nov 26 11:16:24 2019 +0100

Initialize project
Radicle

replication

alice

announce (<project-id>, <ref>, <hash>)

bob

git fetch rad://<project-id>/<ref>
https://radicle.xyz

@cloudhead
https://cloudhead.io