

# Room to grow

Building collaborative, open software

J. Ryan Stinnett

 [convolv.es](https://convolv.es)

# About me

## Firefox

2013 – 2018



Added many features in DevTools, rewrote portions of CSS engine in Rust

## Matrix core team at Element

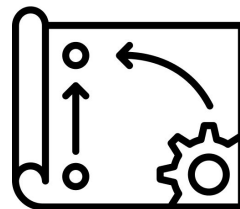
2018 – present



Exploring use cases beyond chat for this open, decentralised, secure communications layer

## Malleable Systems

2019 – present



Exploring how to make programs more malleable and computing more humane

# Outline

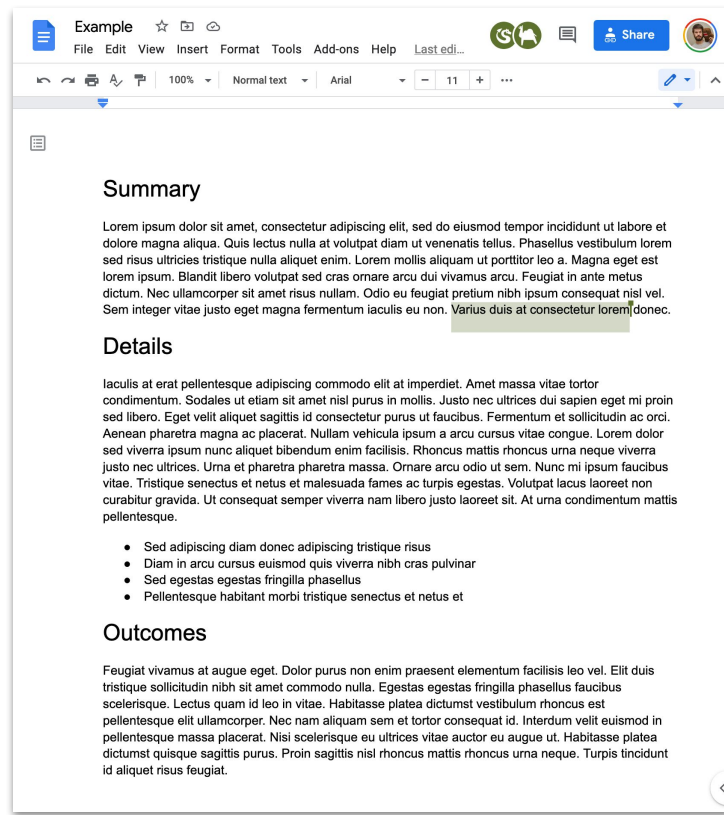
- Collaborative apps, local-first software, data store challenges
- Matrix, what is it, how can it help
- Collaborative documents in Matrix
- Extending towards software generally

# Collaborative apps

- 🎉 Modern creative apps often support collaboration like Google Docs or Figma

but...

- 😓 Most require connectivity to edit
- 👻 Collaborators can watch you all the time
- 😱 No control over your data, could disappear at any time



# Local-first software

If we move the focal point from the server to each user...

- 🕶️ All actions possible locally
- 🙈 Can make changes privately until ready to share again
- 🎉 Your work lives on your machine under your control

Ink & Switch's 2019 essay [Local-first software](#) goes into much more detail, recommended reading!

# Data store challenges

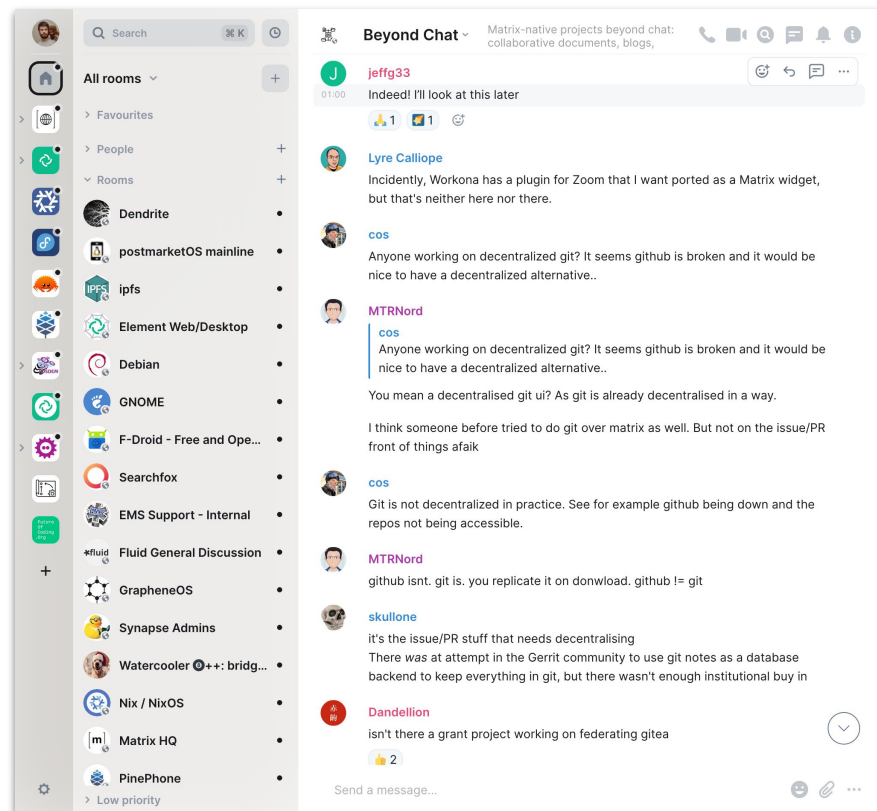
- Local-first software sounds great, but...
- Collaboration algorithm must preserve user intent while resolving disparate edit histories into a single state
- User model needs to support multiple devices per user
- Sharing needs per-user access controls
- Data ideally end-to-end encrypted for privacy
- Building all of the this is a **huge distraction** from actually solving problems for users

# Matrix

- [Matrix](#) is an open standard for interoperable, decentralised, real-time communication over the Internet
- Provides a standard HTTP API for publishing and subscribing to real-time data in specified channels
- General decentralised event store with messaging as a focus
- Supports end-to-end encryption

# Matrix rooms




- Users create and join rooms focused on different topics
- Events in a room form a DAG
- Communities across several rooms organised into spaces
- Each room has its own access controls





# Matrix as a data store

Matrix provides the basic elements we're looking for...

-  Supports multiple devices per user
-  Rooms offer per-user access controls
-  Events are end-to-end encrypted

...which solves a lot of basics, but also...

-  Nothing so far for collaborative data

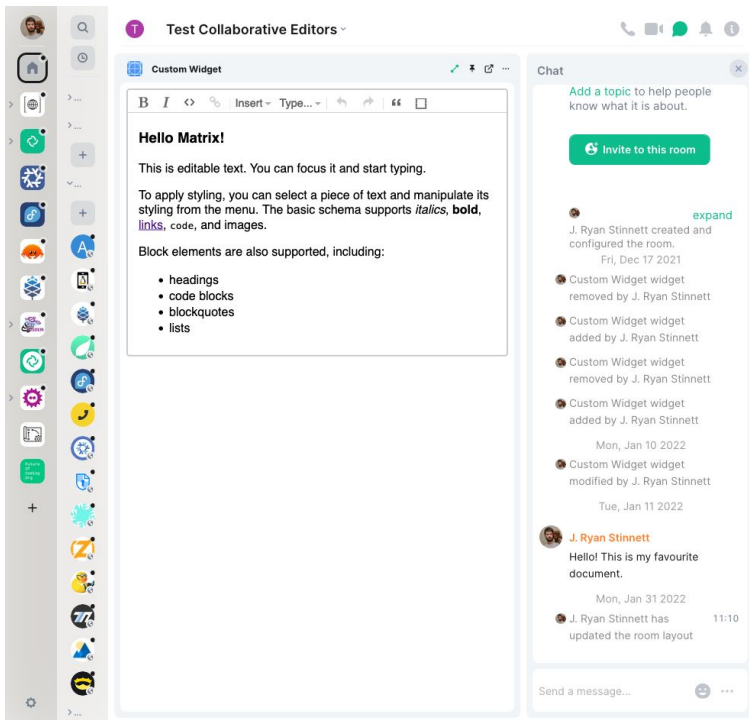
# Documents as rooms

When you have documents stored natively in rooms, you can leverage all the existing Matrix access concepts to invite others, manage permissions, etc. Document access is just the usual access to the room.

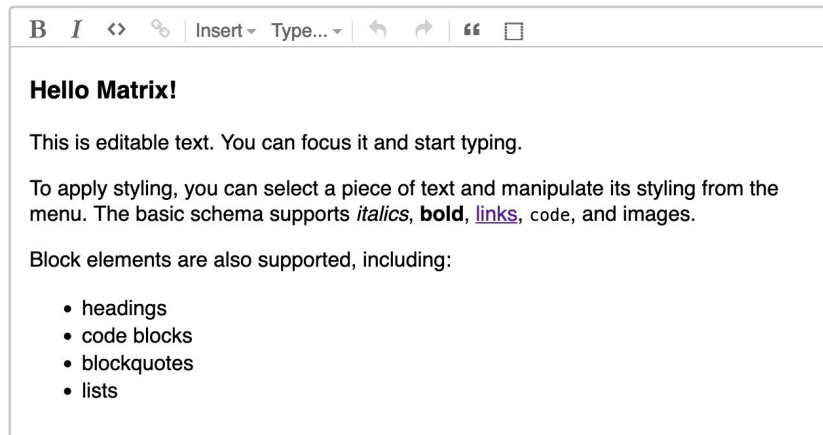
- Room name is the document name
- Spaces provide directories
- Encrypted rooms protect document content
- Invite others to share the document with them
- History visibility controls public viewing and document publishing
- Power levels control read, comment, and edit access
- Chat messages allow for document discussion

# Document editors

Frame within a client



Standalone page



# Events in Matrix

- Events are just JSON
- Matrix spec and various proposals define various event types for messages, files, location sharing, etc.

```
{  
  "content": {  
    "body": "Hello",  
    "msgtype": "m.text",  
  },  
  "type": "m.room.message",  
  "sender": "@jryans:matrix.org",  
  "event_id": "$KVV1yd85tXlBS9z75",  
  "room_id": "!pzVjCQSoQ:matrix.org",  
  "origin_server_ts": 1647739114086,  
}
```

# Document changes as events

- Several recent projects support sending changes via Matrix, but...
- All of these so far are using Matrix as a dumb pipe for binary blobs 🤪
- Those blobs are tied to a single editor stack 😓

```
{  
  "content": {  
    "data": "ewogICJjb250ZW50Ijogd",  
  },  
  "type": "🤪",  
  "sender": "@jryans:matrix.org",  
  "event_id": "$KVV1yd85tXlBS9z75",  
  "room_id": "!pzVjCQSoQ:matrix.org",  
  "origin_server_ts": 1647739114086,  
}
```

# Matrix-native document events

- Human-comprehensible document format, similar to Matrix events themselves
- A document is tree-shaped with associated schema
- Supports an ecosystem of multiple editors accessing the same document
- Detailed proposal in progress

```
{  
  "type": "m.document",  
  "content": {  
    "schema": "#basicmark:example.com@1.0.0",  
    "content": [{  
      "type": "heading",  
      "content": [{  
        "type": "text",  
        "text": "Hello Matrix!",  
      }],  
    }],  
  },  
}
```

# Collaboration algorithms

- Field of collaboration algorithms is still changing rapidly
  - [Peritext](#) for rich text data
- Instead of attempting to bless one, better to describe a general event shape any algorithm could use

```
{
  "type": "m.document.change",
  "content": {
    "algorithm": "com.example.magic",
    "operation": "insert", // algorithm-defined string
    "position": { // algorithm-defined value
      "type": "before",
      "change_id": "<previous event ID>"
    },
    "content": [{ // algorithm-defined value (optional)
      "type": "text",
      "text": "Hello Matrix!"
    }],
  },
}
```

# Programs as rooms

- Store programs (or any tree-shaped data) in rooms
  - JavaScript
  - HTML
  - WebAssembly
- No need to limit to just web platform
  - Rust
  - Clojure
  - Racket
- Collaborative algorithms help here as well to edit programs together
  - See also [Webstrates](#)
- Would be easy to fork, remix, tweak, and share blocks of code



# Summary

- Collaborative, local-first software has many powerful use cases
  - Even if you mainly work alone, still important to have control of your data and ability to share
- Matrix can help with the data store so you can focus on user needs
  - Takes care of users, devices, access controls, E2EE
- Humane events for collaboration data allow for an ecosystem of compatible editors to flourish
  - We can rebuild the user choice and control from traditional desktop apps

# Next steps

- Share proposal for Matrix-native collaborative documents
  - Join the Beyond Chat room (#beyond-chat:matrix.org) to discuss
  - Check the talk page (<https://convolv.es/talks/room-to-grow/>) for updates
- Better permissions via OAuth
- Improved sync protocol to properly handle many, many rooms

# Related work

- Martin Kleppmann, Adam Wiggins, Peter van Hardenberg, Mark McGranaghan  
[Local-first software: You own your data, in spite of the cloud](#) (Apr 2019)
- Geoffrey Litt, Slim Lim, Martin Kleppmann, Peter van Hardenberg  
[Peritext: A CRDT for Rich-Text Collaboration](#) (Nov 2021)
- Matthew Weidner  
[Designing Data Structures for Collaborative Apps](#) (Feb 2022)
- Clemens N. Klokmoose, James R. Eagan, Siemen Baader, Wendy Mackay, Michel Beaudouin-Lafon  
[Webstrates: Shareable Dynamic Media](#) (Nov 2015)

# Thanks!

**J. Ryan Stinnett**



[🏠 convolv.es](https://convolv.es)

[m] @jryans:matrix.org

📧 @jryans@merveilles.town

🐦 @jryans

**Beyond Chat**

2021 – present

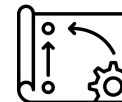


Matrix-native projects beyond chat: collaborative documents, blogs, generic KV databases, tools for thought, etc.

[m] #beyond-chat:matrix.org

**Malleable Systems**

2019 – present



Exploring how to make programs more malleable and computing more humane

[🏠 malleable.systems](https://malleable.systems)

[m] #malleable-systems:matrix.org