Matrix: An open network for secure, decentralized communication

Sumner Evans

August 31, 2021

Beeper

A bit about me

- I graduated in 2018 with my bachelor's in CS from Mines.
- I graduated in 2019 with my master's in CS, also from Mines.
- I worked at The Trade Desk for two years right after graduating.
- I currently am teaching *CSCI 400 Principles of Programming Languages* and I have previously taught *CSCI 406 Algorithms* and *CSCI 564 Advanced Computer Architecture*.
- I started at Beeper in July.

A bit about you

I'd like to get to know everyone a bit more and get a feel for everyone's experience levels.

- How many of you have taken Algorithms?
- How many of you have taken Data Structures?
- How many of you have taken Intro to Computer Science?
- How many of you are in the ACM Matrix chat?

I know you have all had a year of Zoom-class where asking questions is hard, but now there are no excuses: interrupt me at any time if I use a term you don't understand.

A bit about you

I'd like to get to know everyone a bit more and get a feel for everyone's experience levels.

- How many of you have taken Algorithms?
- How many of you have taken Data Structures?
- How many of you have taken Intro to Computer Science?
- How many of you are in the ACM Matrix chat?

I know you have all had a year of Zoom-class where asking questions is hard, but now there are no excuses: interrupt me at any time if I use a term you don't understand.

A bit about you

I'd like to get to know everyone a bit more and get a feel for everyone's experience levels.

- How many of you have taken Algorithms?
- How many of you have taken Data Structures?
- How many of you have taken Intro to Computer Science?
- How many of you are in the ACM Matrix chat?

I know you have all had a year of Zoom-class where asking questions is hard, but now there are no excuses: interrupt me at any time if I use a term you don't understand.

A bit about you

I'd like to get to know everyone a bit more and get a feel for everyone's experience levels.

- How many of you have taken Algorithms?
- How many of you have taken Data Structures?
- How many of you have taken Intro to Computer Science?
- How many of you are in the ACM Matrix chat?

I know you have all had a year of Zoom-class where asking questions is hard, but now there are no excuses: interrupt me at any time if I use a term you don't understand.

A bit about you

I'd like to get to know everyone a bit more and get a feel for everyone's experience levels.

- How many of you have taken Algorithms?
- How many of you have taken Data Structures?
- How many of you have taken Intro to Computer Science?
- How many of you are in the ACM Matrix chat?

I know you have all had a year of Zoom-class where asking questions is hard, but now there are no excuses: interrupt me at any time if I use a term you don't understand.

A bit about you

I'd like to get to know everyone a bit more and get a feel for everyone's experience levels.

- How many of you have taken Algorithms?
- How many of you have taken Data Structures?
- How many of you have taken Intro to Computer Science?
- How many of you are in the ACM Matrix chat?

I know you have all had a year of Zoom-class where asking questions is hard, but now there are no excuses: interrupt me at any time if I use a term you don't understand.

Overview

1. Why Matrix?
2. What does Matrix provide?
3. How does it work?
4. What does Beeper do?
5. Things that I’m excited about in Matrix
6. How to get involved with Matrix
7. A few general tips for everyone

Why Matrix?

Let’s talk about chat platforms…

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Let’s talk about chat platforms…

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Let’s talk about chat platforms…

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Let’s talk about chat platforms…

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Let’s talk about chat platforms…

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Let’s talk about chat platforms…

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Let’s talk about chat platforms…

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Let’s talk about chat platforms…

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Let’s talk about chat platforms…

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Let’s talk about chat platforms…

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Let’s talk about chat platforms…

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Let’s talk about chat platforms…

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Let’s talk about chat platforms…

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Let’s talk about chat platforms…

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Let’s talk about chat platforms…

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Let's talk about chat platforms...

Which of the following chat networks do you use/have you used?

- SMS/MMS
- iMessage
- LinkedIn
- Snapchat
- WhatsApp
- Instagram
- Discord
- Facebook Messenger
- Hangouts
- Slack
- Microsoft Teams
- Signal
- Telegram
- Wire

What do all of these chat networks have in common?

They are **non-interoperable, and many are closed source and/or unencrypted**.

Why is this a problem?

The **closed source** platforms are problematic because you can never be sure *how your data is being used*.

The **unencrypted** platforms are problematic because your messages are not private.

And, because none of them are interoperable, you have to have a ton of chat apps on your phone.

Why is this a problem?

The **closed source** platforms are problematic because you can never be sure *how your data is being used*.

The **unencrypted** platforms are problematic because your messages are not private.

And, because none of them are interoperable, you have to have a ton of chat apps on your phone.

Why is this a problem?

The **closed source** platforms are problematic because you can never be sure *how your data is being used*.

The **unencrypted** platforms are problematic because your messages are not private.

And, because none of them are interoperable, you have to have a ton of chat apps on your phone.

What does Matrix provide?

Matrix solves all your problems

Matrix is an **open** specification for **encrypted**, **decentralized** communication.

It is also designed in such a way that it makes it easy to break down walled garden communication platforms via **bridging**.

Matrix solves all your problems

Matrix is an **open** specification for **encrypted**, **decentralized** communication.

It is also designed in such a way that it makes it easy to break down walled garden communication platforms via **bridging**.

A side note

I first became interested in Matrix when I was the incoming Chair of ACM. Robby (VC) and I tried out most of the open source chat platforms and ended up landing on Matrix because it had all of these characteristics.

Matrix is an *open specification*

Open specifications and standards are all around you. They just make sense™.

Examples:

- Power plugs
- USB
- Wi-Fi
- Every crypto algorithm that's any good

Open protocols allow for *open development* and *clean-room implementations*, they *encourage competition*, and are *externally auditable*.

Matrix is an *open specification*

Open specifications and standards are all around you. They just make sense™.

Examples:

- Power plugs
- USB
- Wi-Fi
- Every crypto algorithm that's any good

Open protocols allow for *open development* and *clean-room implementations*, they *encourage competition*, and are *externally auditable*.

Matrix is an *open specification*

Open specifications and standards are all around you. They just make sense™.

Examples:

- Power plugs
- USB
- Wi-Fi
- Every crypto algorithm that's any good

Open protocols allow for *open development* and *clean-room implementations*, they *encourage competition*, and are *externally auditable*.

Matrix is *encrypted* by default*

Matrix has encryption built-in. It is implemented using Olm, which is a clone of the Signal protocol

Matrix is *decentralized*

The Matrix architecture is actually a *federated* architecture.

Individual devices communicate to a *homeserver* which anyone can host.

The homeserver communicates with other homeservers in the federation.

Think of it like email. You can email somebody using Outlook from Gmail.*

Every server in the federation gets a copy of a room, so no one entity controls the network.

This also means that the network is resilient to individual server outages, or even wider internet outages.

Matrix is *decentralized*

The Matrix architecture is actually a *federated* architecture.

Individual devices communicate to a *homeserver* which anyone can host.

The homeserver communicates with other homeservers in the federation.

Think of it like email. You can email somebody using Outlook from Gmail.*

Every server in the federation gets a copy of a room, so no one entity controls the network.

This also means that the network is resilient to individual server outages, or even wider internet outages.

Matrix is *decentralized*

The Matrix architecture is actually a *federated* architecture.

Individual devices communicate to a *homeserver* which anyone can host.

The homeserver communicates with other homeservers in the federation.

Think of it like email. You can email somebody using Outlook from Gmail.*

Every server in the federation gets a copy of a room, so no one entity controls the network.

This also means that the network is resilient to individual server outages, or even wider internet outages.

Matrix is *decentralized*

The Matrix architecture is actually a *federated* architecture.

Individual devices communicate to a *homeserver* which anyone can host.

The homeserver communicates with other homeservers in the federation.

Think of it like email. You can email somebody using Outlook from Gmail.*

Every server in the federation gets a copy of a room, so no one entity controls the network.

This also means that the network is resilient to individual server outages, or even wider internet outages.

Matrix allows for *bridges* and *bots*

Bridges bring external chat networks into Matrix. More on this later.

Bots allow for automated interactions and notifications.

Matrix allows for *bridges* and *bots*

Bridges bring external chat networks into Matrix. More on this later.

Bots allow for automated interactions and notifications.

How does it work?

Architecture

Every server has a copy of the room, but how do we keep that in sync?

The architecture of Matrix does this in a way that ensures *eventual consistency*.

Even if the server where the room was created goes down, people can still communicate.

When a broken server comes back online, it will receive all the *events* (messages).

Let's look at the animation on Matrix.org...

Architecture

Every server has a copy of the room, but how do we keep that in sync?

The architecture of Matrix does this in a way that ensures *eventual consistency*.

Even if the server where the room was created goes down, people can still communicate.

When a broken server comes back online, it will receive all the *events* (messages).

Let's look at the animation on Matrix.org...

Architecture

Every server has a copy of the room, but how do we keep that in sync?

The architecture of Matrix does this in a way that ensures *eventual consistency*.

Even if the server where the room was created goes down, people can still communicate.

When a broken server comes back online, it will receive all the *events* (messages).

Let's look at the animation on Matrix.org...

Architecture

Every server has a copy of the room, but how do we keep that in sync?

The architecture of Matrix does this in a way that ensures *eventual consistency*.

Even if the server where the room was created goes down, people can still communicate.

When a broken server comes back online, it will receive all the *events* (messages).

Let's look at the animation on Matrix.org...

Client-Server API

The **Client-Server API** specifies how clients communicate with their homeserver.

Demo!

Client-Server API

The **Client-Server API** specifies how clients communicate with their homeserver.

Demo!

Federation (Server-Server) API

The **Server-Server API** or **Federation API** specifies how servers communicate with other servers to ensure that everyone has the same room state.

A bit of graph theory

- A **graph** is a collection of *nodes* connected by *edges*.
- A **directed graph** is a graph where the edges are *directional* (have arrows).
- An **acyclic graph** is a graph that has no cycles/loops.
- A **directed acyclic graph (DAG)** is a directional graph with no cycles.

A bit of graph theory

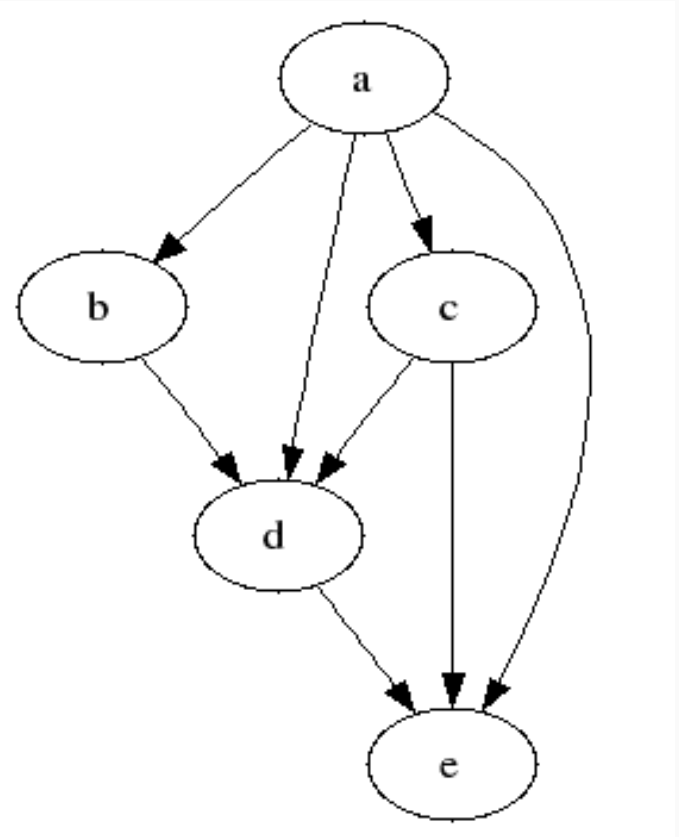
- A **graph** is a collection of *nodes* connected by *edges*.
- A **directed graph** is a graph where the edges are *directional* (have arrows).
- An **acyclic graph** is a graph that has no cycles/loops.
- A **directed acyclic graph (DAG)** is a directional graph with no cycles.

A bit of graph theory

- A **graph** is a collection of *nodes* connected by *edges*.
- A **directed graph** is a graph where the edges are *directional* (have arrows).
- An **acyclic graph** is a graph that has no cycles/loops.
- A **directed acyclic graph (DAG)** is a directional graph with no cycles.

A bit of graph theory

- A **graph** is a collection of *nodes* connected by *edges*.
- A **directed graph** is a graph where the edges are *directional* (have arrows).
- An **acyclic graph** is a graph that has no cycles/loops.
- A **directed acyclic graph (DAG)** is a directional graph with no cycles.



The event DAG

Matrix rooms are represented by a DAG of *events* representing things such as messages, joins, leaves, etc.

The DAG provides a *partial ordering* of events in the room because every event has zero or more "parent" events.

This is similar to Git where every commit has 0 or more "parent" commits.

See `https://matrix.org/docs/spec/#event-graphs`

The event DAG

Matrix rooms are represented by a DAG of *events* representing things such as messages, joins, leaves, etc.

The DAG provides a *partial ordering* of events in the room because every event has zero or more “parent” events.

This is similar to Git where every commit has 0 or more “parent” commits.

See `https://matrix.org/docs/spec/#event-graphs`

The event DAG

Matrix rooms are represented by a DAG of *events* representing things such as messages, joins, leaves, etc.

The DAG provides a *partial ordering* of events in the room because every event has zero or more “parent” events.

This is similar to Git where every commit has 0 or more “parent” commits.

See `https://matrix.org/docs/spec/#event-graphs`

Event types

There are two main event types: **message events** and **state events**.

Message events:
These describe transient 'once-off' activity in a room such as an instant messages, VoIP call setups, file transfers, etc. They generally describe communication activity.

State events:
These describe updates to a given piece of persistent information ('state') related to a room, such as the room's name, topic, membership, participating servers, etc. State is modelled as a lookup table of key/value pairs per room, with each key being a tuple of `state_key` and `event type`. Each state event updates the value of a given key.

See `https://matrix.org/docs/spec/#room-structure`

Event types

There are two main event types: **message events** and **state events**.

Message events:
These describe transient 'once-off' activity in a room such as an instant messages, VoIP call setups, file transfers, etc. They generally describe communication activity.

State events:
These describe updates to a given piece of persistent information ('state') related to a room, such as the room's name, topic, membership, participating servers, etc. State is modelled as a lookup table of key/value pairs per room, with each key being a tuple of `state_key` and `event type`. Each state event updates the value of a given key.

See `https://matrix.org/docs/spec/#room-structure`

Event types

There are two main event types: **message events** and **state events**.

Message events:
These describe transient 'once-off' activity in a room such as an instant messages, VoIP call setups, file transfers, etc. They generally describe communication activity.

State events:
These describe updates to a given piece of persistent information ('state') related to a room, such as the room's name, topic, membership, participating servers, etc. State is modelled as a lookup table of key/value pairs per room, with each key being a tuple of `state_key` and `event type`. Each state event updates the value of a given key.

See `https://matrix.org/docs/spec/#room-structure`

What does Beeper do?

Things that I’m excited about in Matrix

Bridges

I work on the bridge team at Beeper, so I am obviously excited about the possibilities there.

I hope that in the future, it will be a no-brainer to base all new chat applications on top of Matrix.

I also hope that existing chat networks rebase on Matrix, so bridges become unnecessary.

Bridges

I work on the bridge team at Beeper, so I am obviously excited about the possibilities there.

I hope that in the future, it will be a no-brainer to base all new chat applications on top of Matrix.

I also hope that existing chat networks rebase on Matrix, so bridges become unnecessary.

Bots

I'm excited about the possibilities for more interactivity bots.

Right now, basically only IRC-style text-based communication is possible with bots.

There are MSCs for things like buttons, suggested replies, etc. which will really make bots more featureful.

Bots

I’m excited about the possibilities for more interactivity bots.

Right now, basically only IRC-style text-based communication is possible with bots.

There are MSCs for things like buttons, suggested replies, etc. which will really make bots more featureful.

Non-chat applications

Many people are experimenting with trying to use Matrix for things other than chat.

- Social media (Cerulean and Circles)
- Blogs
- Comment systems for blogs
- Collaborative notepads
- Collaborative whiteboards

All of these applications take advantage of the fact that at the end of the day, Matrix is just a distributed DAG.

Client features

I’m excited about a bunch of features that are coming soon™ including:

- Spaces and the many features that it will bring that will help facilitate community management
- Threads
- Extensible profiles (profiles as rooms)

How to get involved with Matrix

Right here at Mines

You can get involved in Matrix right here at ACM!

- You can join the ACM chat!
- If you are interested in building non-chat applications on top of Matrix, consider joining Visplay.

Follow the news

The Matrix community is very open. Most of the development happens in the open, and you can join development and support chat rooms.

Here are a few to get you started:

- Matrix News: `#matrix-news:matrix.org`
- This Week in Matrix (TWIM): `#twim:matrix.org`
- Matrix Community Space: `#community:matrix.org` (and subspaces)

Most Matrix projects also have their own chat room.

Run Synapse

You can run your own homeserver. Synapse is the most featureful server, and is relatively easy to set up.

You can register a domain name for free for a year with the GitHub Student Pack.

Contribute documentation or code

There are many projects which need more documentation or code. The Matrix Community space and its subspaces are a good place to find projects.

- There are many existing projects that you can contribute to in the Matrix ecosystem: Element client codebases, Synapse, Dendrite, a bunch of bots and bridges.
- You can even contribute to the Matrix spec itself via MSCs.
- Write a new bridge (Beeper may even pay you to do that, contact Eric `@eric:beeper.com`)
- Write bots
- Consider joining Beeper! Contact Eric (`@eric:beeper.com`) if you are interested.

Contribute documentation or code

There are many projects which need more documentation or code. The Matrix Community space and its subspaces are a good place to find projects.

- There are many existing projects that you can contribute to in the Matrix ecosystem: Element client codebases, Synapse, Dendrite, a bunch of bots and bridges.
- You can even contribute to the Matrix spec itself via MSCs.
- Write a new bridge (Beeper may even pay you to do that, contact Eric `@eric:beeper.com`)
- Write bots
- Consider joining Beeper! Contact Eric (`@eric:beeper.com`) if you are interested.

Contribute documentation or code

There are many projects which need more documentation or code. The Matrix Community space and its subspaces are a good place to find projects.

- There are many existing projects that you can contribute to in the Matrix ecosystem: Element client codebases, Synapse, Dendrite, a bunch of bots and bridges.
- You can even contribute to the Matrix spec itself via MSCs.
- Write a new bridge (Beeper may even pay you to do that, contact Eric `@eric:beeper.com`)
- Write bots
- Consider joining Beeper! Contact Eric (`@eric:beeper.com`) if you are interested.

Contribute documentation or code

There are many projects which need more documentation or code. The Matrix Community space and its subspaces are a good place to find projects.

- There are many existing projects that you can contribute to in the Matrix ecosystem: Element client codebases, Synapse, Dendrite, a bunch of bots and bridges.
- You can even contribute to the Matrix spec itself via MSCs.
- Write a new bridge (Beeper may even pay you to do that, contact Eric `@eric:beeper.com`)
- Write bots
- Consider joining Beeper! Contact Eric (`@eric:beeper.com`) if you are interested.

Contribute documentation or code

There are many projects which need more documentation or code. The Matrix Community space and its subspaces are a good place to find projects.

- There are many existing projects that you can contribute to in the Matrix ecosystem: Element client codebases, Synapse, Dendrite, a bunch of bots and bridges.
- You can even contribute to the Matrix spec itself via MSCs.
- Write a new bridge (Beeper may even pay you to do that, contact Eric `@eric:beeper.com`)
- Write bots
- Consider joining Beeper! Contact Eric (`@eric:beeper.com`) if you are interested.

A few general tips for everyone

Do something

Don’t be paralysed because you don’t know what you want to do. **Just do something.**

If you don’t know what interests you, try things until you find something that you are sufficiently motivated by.

If you don’t like what you are doing, you can always get a different job.

Getting into open source

If you want to become and open source developer, start by being a good open source project user.

Then start reporting issues and submitting documentation fixes.

Then maybe you can even start answering questions that people ask on chat rooms or GitHub issues associated with the project.

Then make small code fixes.

Anyone can contribute to open source, you just have to have realistic expectations about what that means.

Getting into open source

If you want to become and open source developer, start by being a good open source project user.

Then start reporting issues and submitting documentation fixes.

Then maybe you can even start answering questions that people ask on chat rooms or GitHub issues associated with the project.

Then make small code fixes.

Anyone can contribute to open source, you just have to have realistic expectations about what that means.

Getting into open source

If you want to become and open source developer, start by being a good open source project user.

Then start reporting issues and submitting documentation fixes.

Then maybe you can even start answering questions that people ask on chat rooms or GitHub issues associated with the project.

Then make small code fixes.

Anyone can contribute to open source, you just have to have realistic expectations about what that means.

Getting into open source

If you want to become and open source developer, start by being a good open source project user.

Then start reporting issues and submitting documentation fixes.

Then maybe you can even start answering questions that people ask on chat rooms or GitHub issues associated with the project.

Then make small code fixes.

Anyone can contribute to open source, you just have to have realistic expectations about what that means.

Getting into open source

If you want to become and open source developer, start by being a good open source project user.

Then start reporting issues and submitting documentation fixes.

Then maybe you can even start answering questions that people ask on chat rooms or GitHub issues associated with the project.

Then make small code fixes.

Anyone can contribute to open source, you just have to have realistic expectations about what that means.