Matrix in the French State

and introducing…

Matrix 1.0

matthew@matrix.org
@matrixdotorg

Matrix is an open network for secure, decentralised real-time communication.

Interoperable chat

Interoperable VoIP

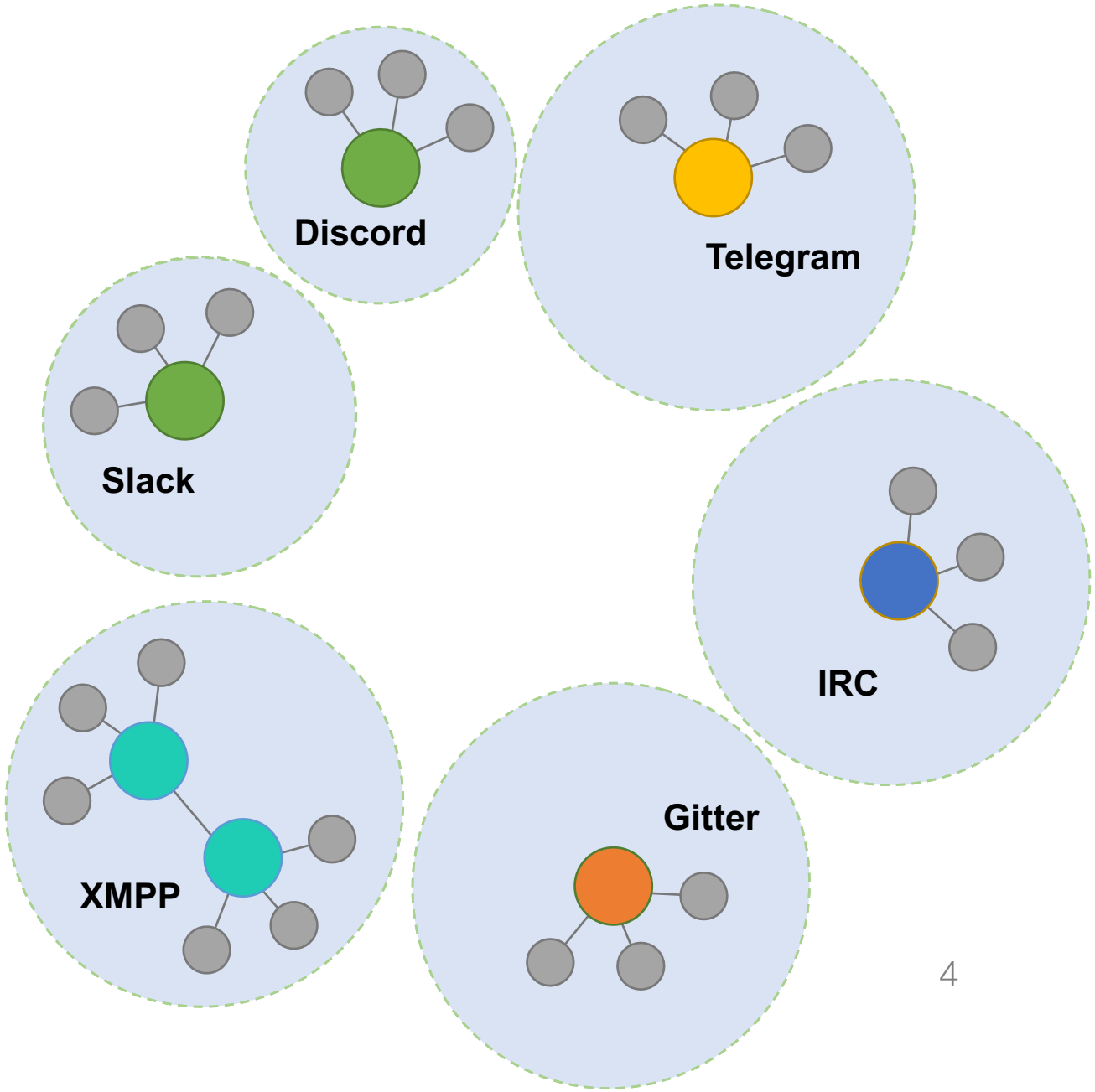
Open comms for VR/AR

Real-time IoT data fabric

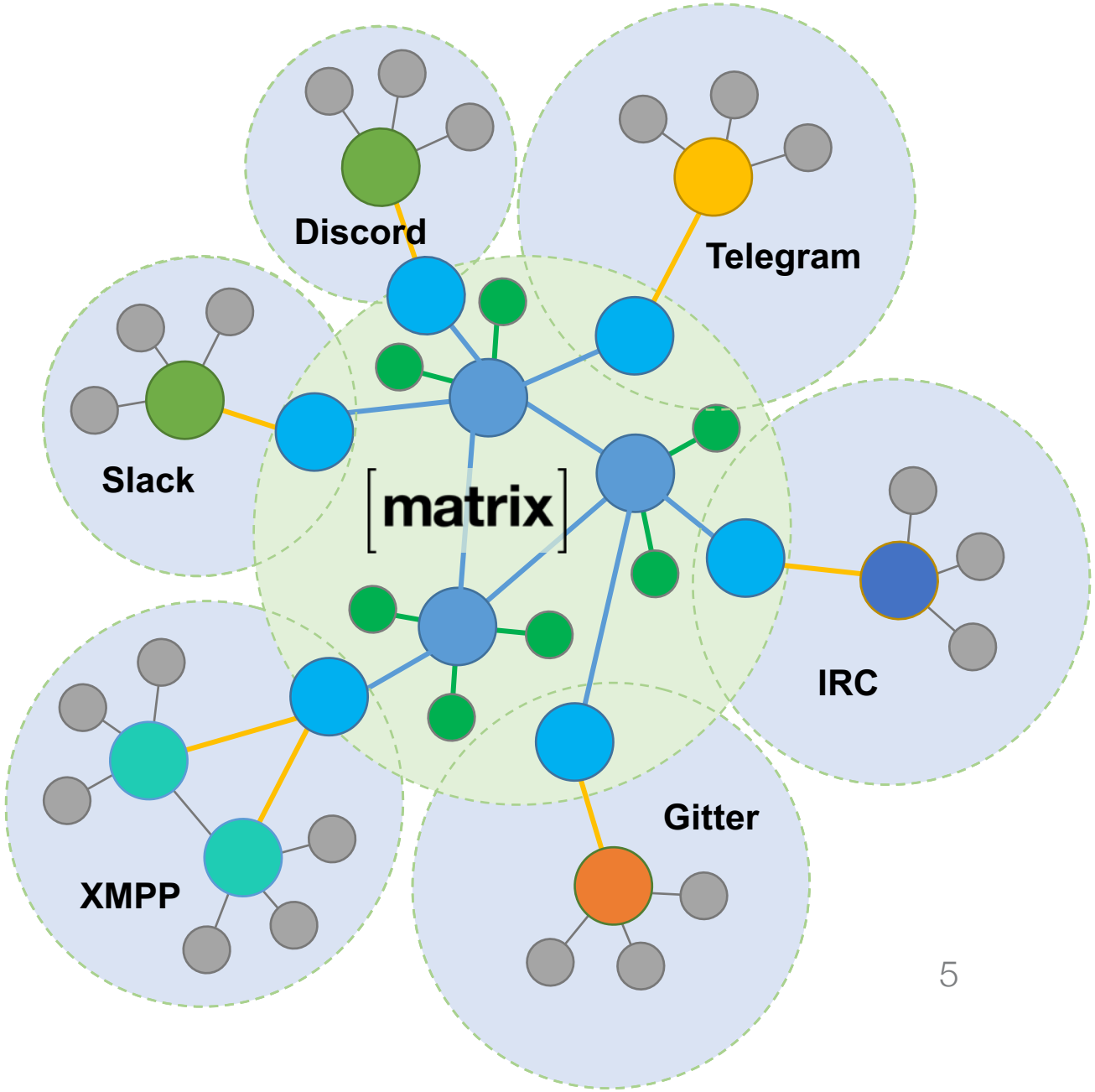
matrix

Mission: to create a global decentralised encrypted comms network that provides an open platform for real-time communication.

matrix



[matrix]



[matrix]

No single party owns your conversations.

Conversations are shared over all participants.

Matrix Architecture

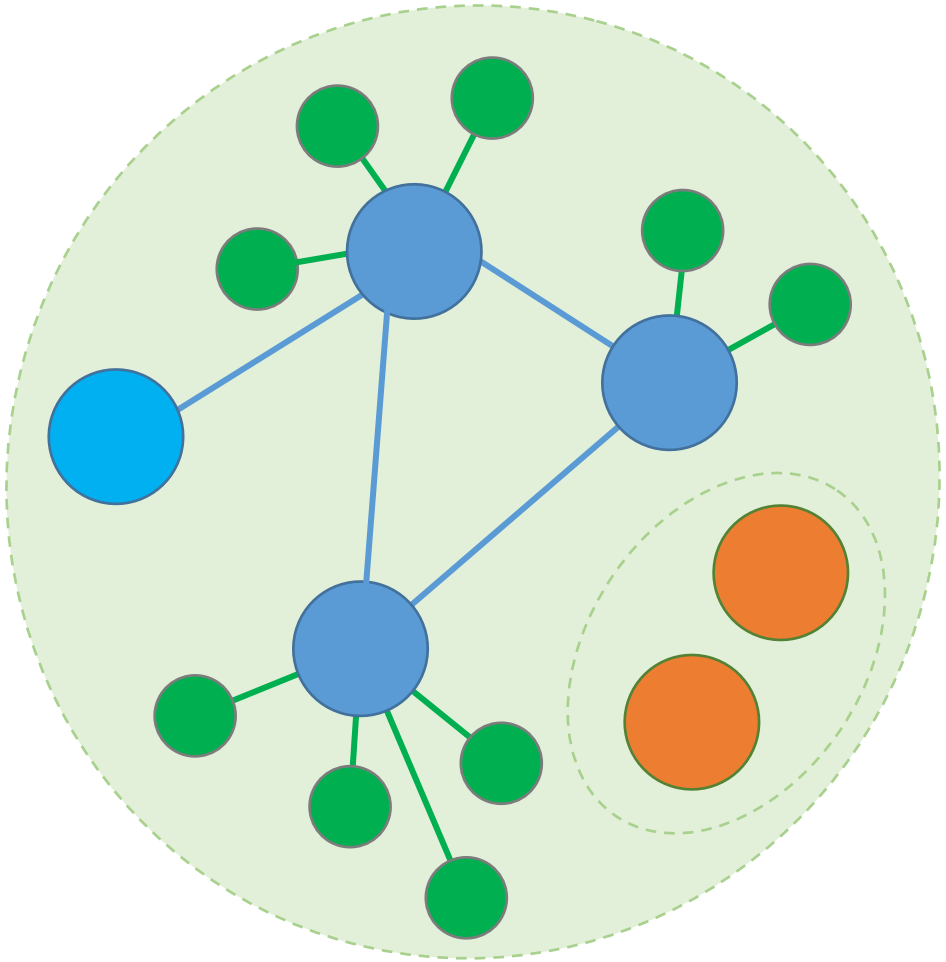
matrix

Clients

Home Servers

Application Servers

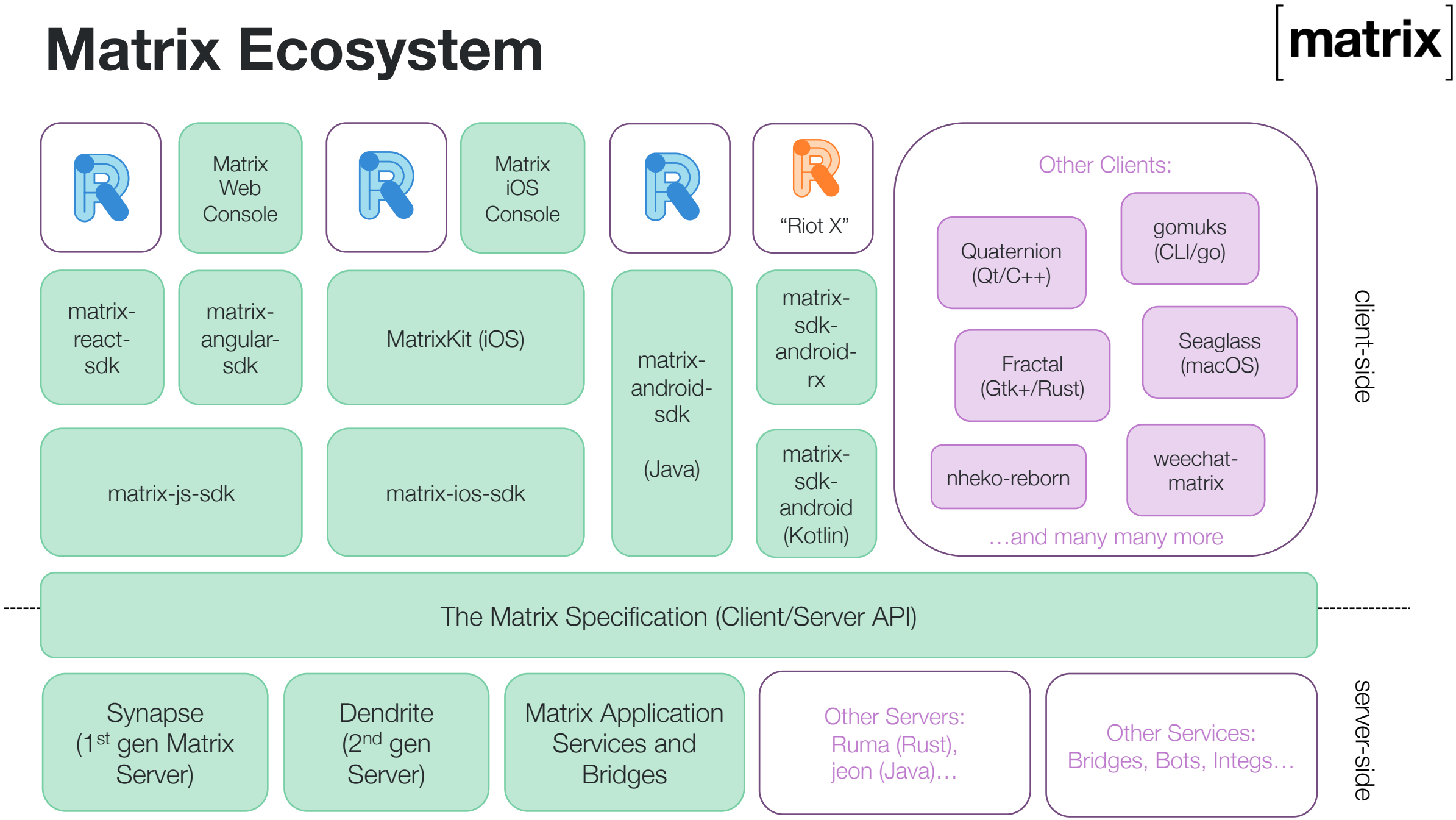
Identity Servers



What do you get in the spec?

- Decentralised **conversation history**
- **Group** Messaging (and 1:1)
- **End-to-end Encryption**
- **VoIP** signalling for WebRTC
- Server-side **push notification** rules
- Server-side **search**
- Read receipts, Typing Notifs, Presence
- Synchronised read state and **unread counts**
- Decentralised **content repository**
- “**Account data**” for users per room

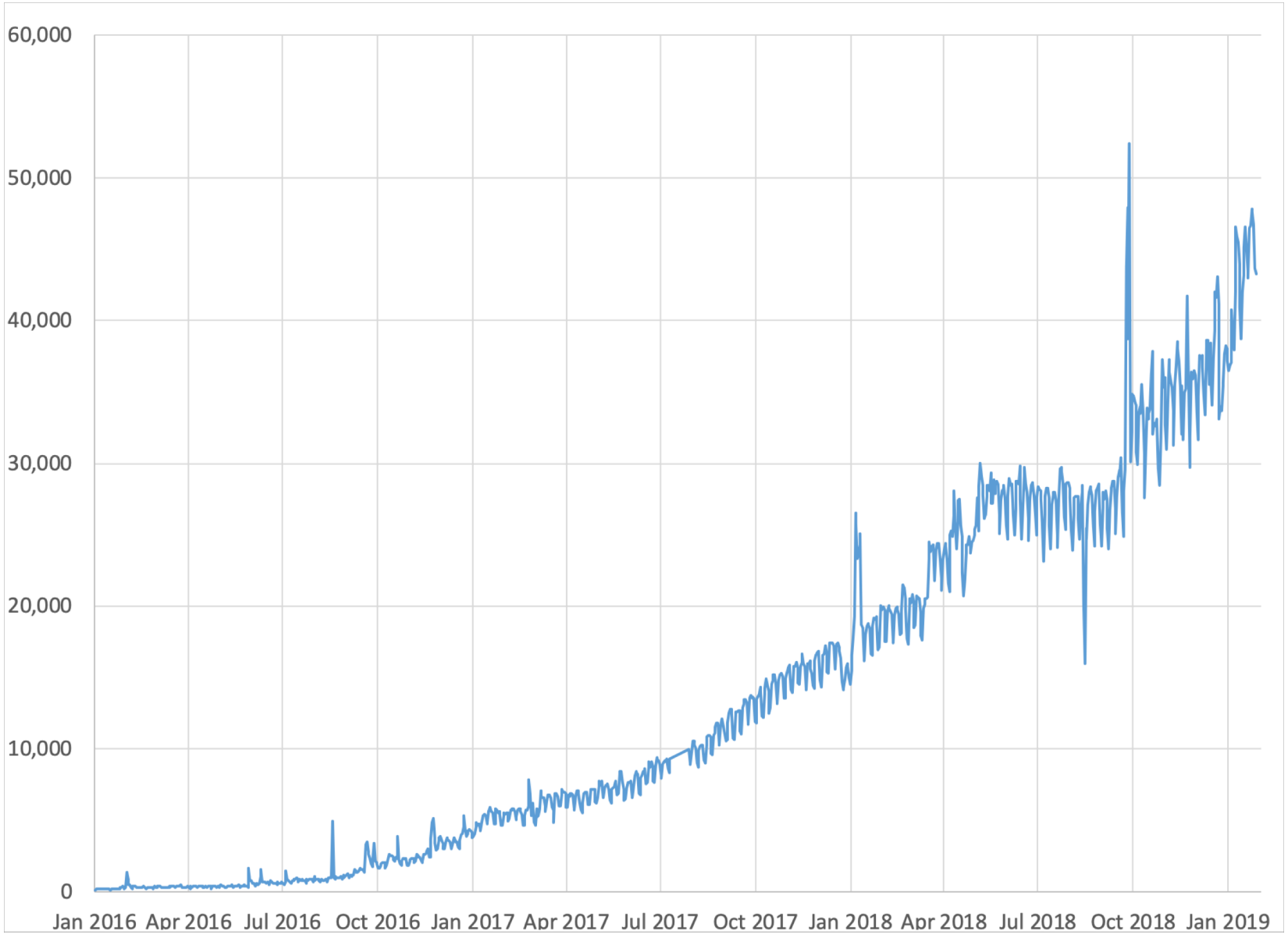
Matrix Ecosystem



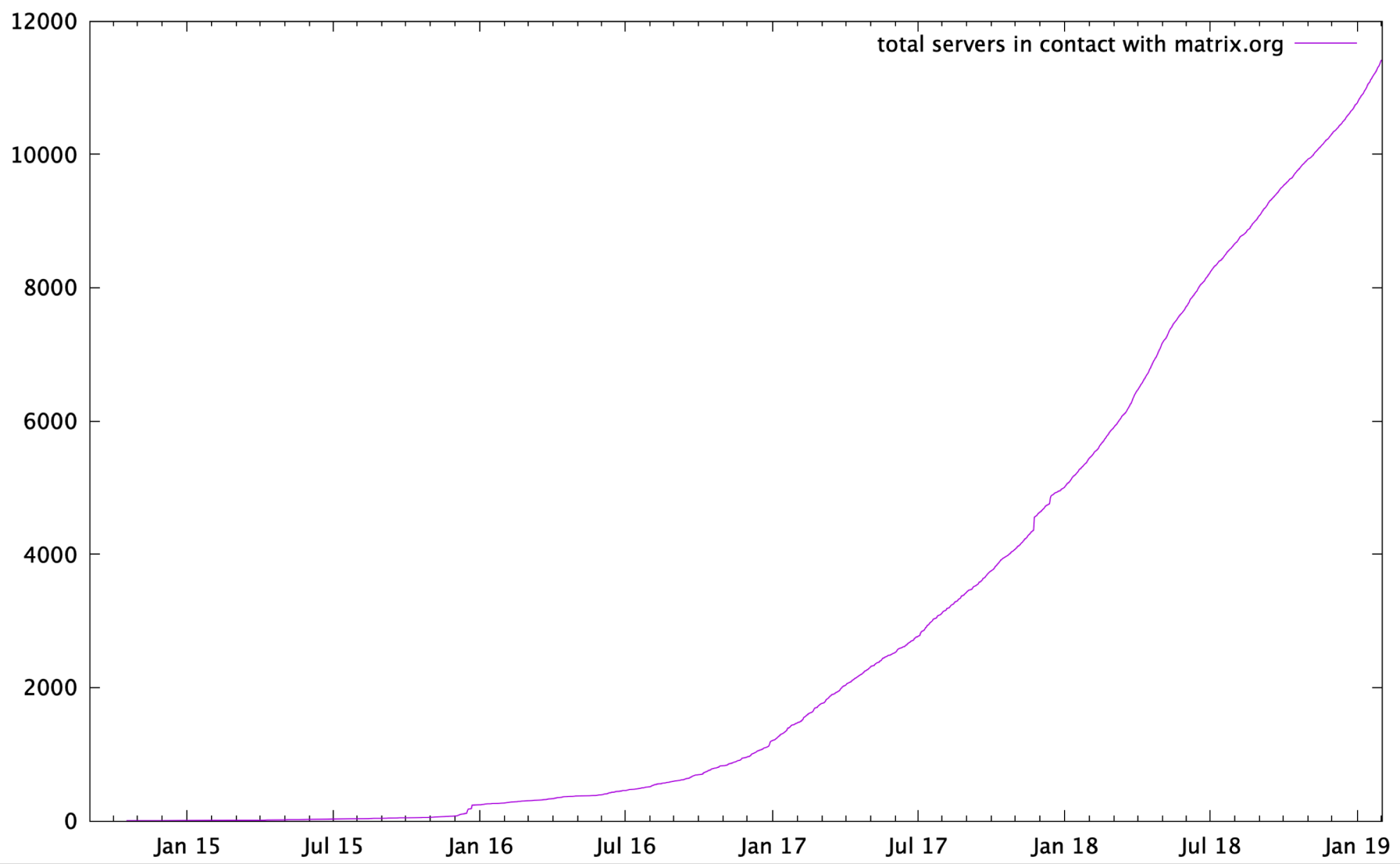
A brief history of Matrix

- 2014: First alpha!
- 2015: Federation becomes usable; add Postgres; add IRC
- 2015: First release of Vector as a flagship Matrix client; r0 CS API
- 2016: Scaling; First cut of E2E Encryption; Vector becomes Riot
- 2017: Widgets, Stickers, Jitsi, Communities, i18n, Dendrite,
- 2018: Feature freeze. Road to 1.0: security, stability, governance.
- 2019: **Matrix 1.0** and beyond!

Daily active users on matrix.org



Publicly visible servers



Community Status

- ~7.0M global visible accounts
- ~2.0M messages per day
- ~1.5M unbridged accounts
- ~1.5M messages per day
- ~1.5M rooms that Matrix.org participates in
- ~12,000 federated servers
- ~2000 msgs/s out, ~20 msgs/s in on Matrix.org
- ~300 projects building on Matrix
- ~60 companies building on Matrix

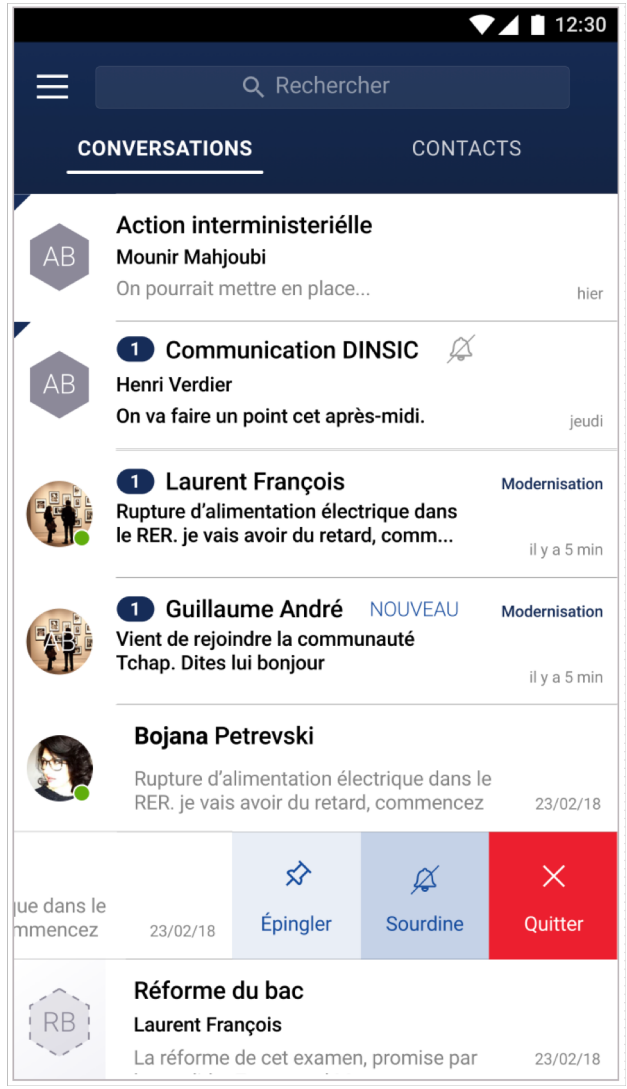
What about France?

Matrix in France

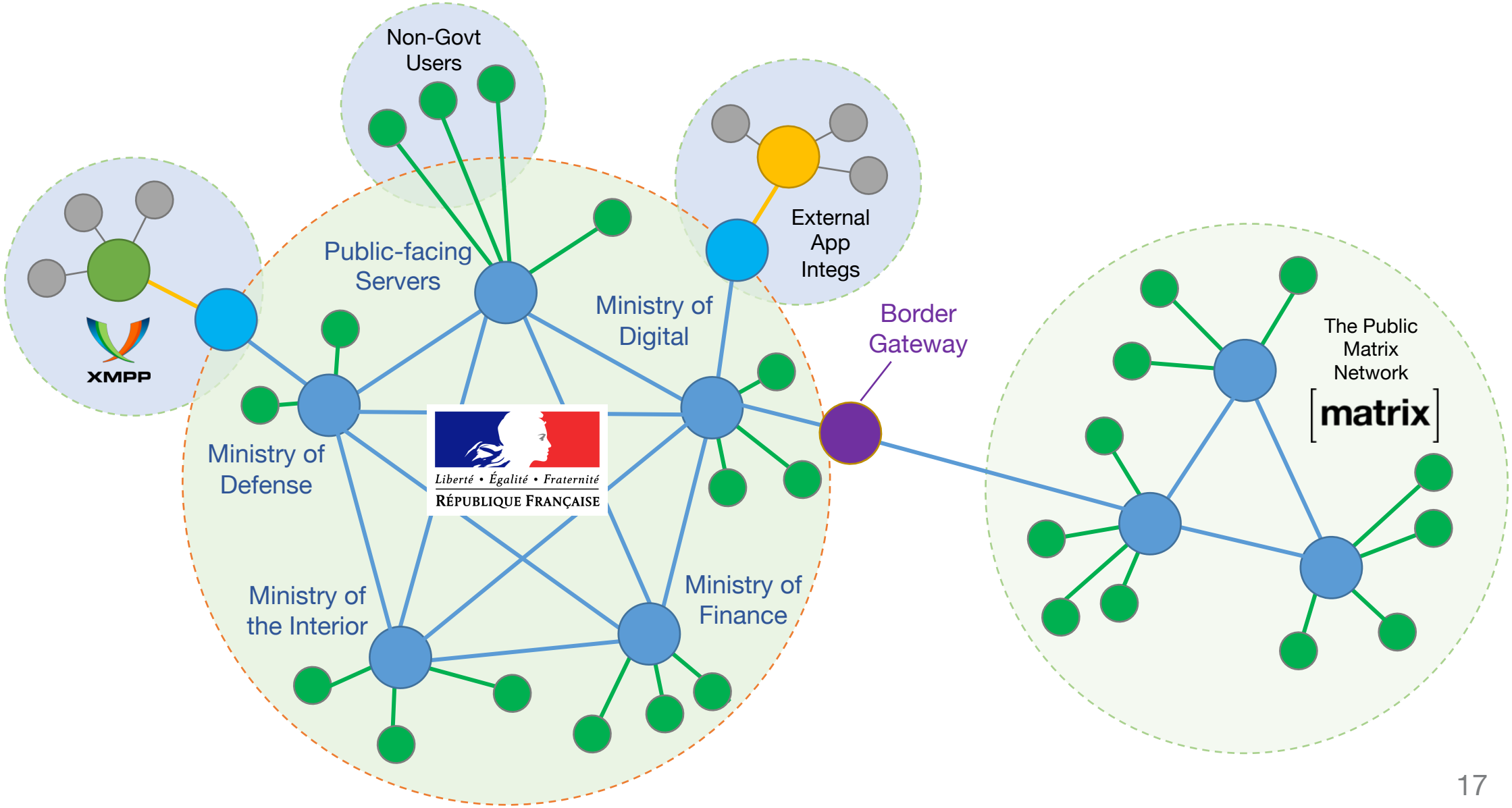
- DINSIC (French Ministry of Digital) reached out at beginning of 2018 to ask about using Matrix to provide self-sovereign, encrypted, decentralised communication across the government.
- Requirements:
  - 100% Open source.
  - >30 operationally independent deployments (per-Ministry granularity)
  - 5.5 Million users.
  - Mandatory E2E Encryption.
  - Enterprise-grade antivirus support.
  - Different security zones.
  - Initially private federation, but with scope to support public federation(!)

Current Status

- Dev started on the app in May
- Apache-licensed fork of Riot
  - https://github.com/dinsic-pim
- Android first, then Web, then iOS
- Trial started in June
- Audits from ANSSI and EY
- As of Jan, rolling out across all the ministries
- Lots and lots of Ansible
- **Demo!**



Very Approximate Schematic (not remotely accurate)



Driven by France: E2E-capable Antivirus

- No choice but exfiltrate keys for files to a scanner service.
- We do this as optimally as possible.
- We send the service only the target file URL and the encryption keys for the file (**not** the Matrix event or room) – in turn encrypted to the service's pinned public key.
- The service is completely operationally isolated from the homeserver
- All media uploads & downloads are then proxied via HTTPS to an ICAP-based scanning service
- To be added to the Spec!
- https://github.com/matrix-org/matrix-content-scanner

Driven by France: Infra automation via Ansible

- Lots and lots of ansible to automates all the config management across all the deployments.
- 27 Roles including:
  - matrix-synapse-dinsic
  - matrix-sydent-dinsic
  - matrix-postgres-dinsic
  - matrix-content-scanner-dinsic
  - matrix-rageshake-dinsic
  - matrix-coturn-dinsic
  - etc. etc.
- Not FOSS, alas, but based on https://github.com/matrix-org/matrix-ansible-synapse etc

```
ansible matthew$ find . | wc -l
     730
```

Driven by France: Performance

- Lazy Loading Members
  - Option now to only sync membership data to clients for the members currently speaking in a room.
  - Saves 3-5x bandwidth and RAM on the client.
- Python 3
  - As of 0.34, Synapse runs on Python 3!
  - Big RAM savings thanks to UTF32 -> UTF8 string representation
  - Surprisingly large CPU savings for some workloads; noticeably snappier
- Lots of incremental operational work on Synapse to profile bottlenecks and improve its caching.

Driven by France (well, sort of):

Matrix 1.0

Matrix 1.0

- Cut stable spec releases of **all** Matrix APIs
- “First we make it correct, then we make it fast”
- Fix design issues in the Server Server API
- Add the infrastructure to migrate between protocol versions
- Finalise The Matrix.org Foundation governance
- Land remaining work needed to turn on E2E encryption by default
- …and then exit Beta at last!

Matrix 1.0: Server Server API r0.1

- We have been (very) slow on a stable spec for the Federation API.
- Client/Server API r0.1 was released back in Oct 2015
- Identity, Application & Push API r0.1s landed in Aug 2018
- But the Federation (Server/Server) API had some design flaws:
  - We chose Perspectives for key management between servers rather than X.509 CAs. This was before Lets Encrypt, however…
  - The merge resolution algorithm for room state could yield unexpected results (e.g. merging a stale copy of the room could cause the state of the room to reset to the older state, a so called “State Reset”).
  - Servers could select arbitrary event IDs… meaning malicious servers can create events with deliberately clashing IDs.
  - We didn’t provide from the outset the ability to version rooms in order to upgrade their protocol.
- **These are now fixed – SS API r0.1 was released last night! (Feb 1 2019)**

Matrix 1.0: No More Self-signed Certificates.

- Historically Matrix used Perspective notaries for tracking trust of TLS certificates and Matrix signing keys.
- Like TOFU, but with consensus within a room.
- Speeds up setup as you don't have to get a cert from a CA!
- In practice, we didn't finish the consensus work, so almost everyone used the default Matrix.org homeserver as the default notary => Centralisation ☹
- Meanwhile, Perspectives seems to be dead.

As of Matrix 1.0 (SS r0.1), we require homeservers to present a CA-signed TLS certificate.

Synapse 0.99 natively speaks ACME and will autogenerate you one from Lets Encrypt! Please delete your self-signed certs!

Synapse 1.0 (Mar 2019) will refuse to connect to self-signed servers.

Matrix 1.0: .well-known for SS discovery

- A side-effect of requiring genuine TLS certificates means that if bigcorp.com wants to run a Matrix server, they might need to give whoever runs the server a TLS cert for the bigcorp.com.
- This is a big problem if the server is being outsourced.
- You could use SRV to delegate to bigcorp.matrixhosting.com, but trusting whatever certificate the SRV points to is quite vulnerable to DNS poisoning.
- So instead, we have added .well-known/matrix/server to let bigcorp.com delegate to a given server.
- SRV and TLS validation is then applied to the delegated server as normal.

If you are running a server for domain you can't generate a TLS certificate for, please add a .well-known URI to delegate it to a hostname you can control. (Supported in Synapse 0.99)

Matrix 1.0: State Resolution Reloaded

- In SS API r0.1 we have completely changed the State Resolution algorithm used to merge rooms together.
- It's a Hard Problem as:
  - Rooms are made up of a Directed Acyclic Graph of messages.
  - For performance, we don't want to force all servers to compare a full copy of the DAG, or even a full copy of the 'state events' in a DAG.
  - We now look at the "auth events" in a room to apply an ordering when merging the rooms and prevent unexpected results.
  - We also no longer consider the untrustable "depth" metric.

Matrix 1.0 supports versioning of rooms, so rooms can be upgraded to the new resolution algorithm ("Version 2" rooms)

Matrix 1.0: Event IDs as Hashes

- In SS API r0.1 we have also completely changed the shape of Event IDs.
- Before: `$1549058849131269kpYEr:matrix.org`
- After: `$zc4ip/DpPI9FZVLM1wN9RLqN19vuVBURmIqAohZ1HXg`
  - Base64 hash of the message contents.

“Version 3” rooms in SS API r0.1 support the new ID shape.

We will be encouraging everyone to upgrade to V3 rooms once the majority of the network supports them (i.e. runs Synapse 0.99 or later or equivalent)

Matrix 1.0: The Matrix.org Foundation

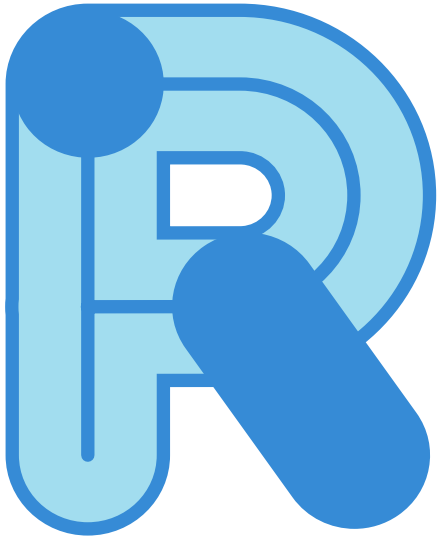
- We have finalised the Open Governance model for Matrix.org
- Matrix Spec Change 1779
- Not-for-profit UK Community Interest Company
- Neutral custodian of the standard.
- ~8 person Spec Core Team
- ~5 person Guardians (Foundation Board of Directors)
- Currently turning MSC1779 into Articles of Association and finalising the Guardians.

Matrix 1.0: Timeline

- SS API r0.1 released Feb 1st 2019
- First time we've ever had stable releases of all APIs!
- However, need to release CS API r0.5 and possibly IS API r0.2
- …and then we will declare the end result Matrix 1.0 and exit beta!
- Synapse 0.99 speaks SS API r0.1.
  - You can upgrade to the final(?) release candidate 0.99.0rc4 today.
- **Synapse 1.0 will drop compatibility with pre-r0.1 SS API, so will follow after ~1 month in March 2019.**

Riot Redesign!

[matrix]



RIOT.IM

https://riot.im/develop

E2E Encryption

Olm: What's next?

- Ability to share session ratchet data with new devices or new room participants
- Cross-signing device keys?
- Better device verification
- Better push notification UX for E2E rooms
- Better primitives & performance
- Turning on E2E by default for rooms with private history
- Negotiating E2E with legacy clients(?)

Olm: What's next?

- Ability to share session ratchet data with new devices or new room participants ✅
- Cross-signing device keys? ✅
- Better device verification ✅
- Key recovery backups ✅
- Better push notification UX for E2E rooms ✅
- Better primitives & performance ✅
- E2E-capable Search
- Turning on E2E by default for rooms with private history
- Negotiating E2E with legacy clients(?)

Other New Stuff!

Other new stuff!

- “RiotX” on Android
  - Ground-up rewrite in Kotlin and RX
  - Roughly 10x faster than Riot/Android
  - Implements the redesign look & feel!
- Bifröst – One (Rainbow) Bridge to Rule Them All
  - Bridge app with pluggable backends:
    - XMPP.js for high quality XMPP bridging:
      - connects from Matrix to XMPP MUCs and PMs
      - …and from XMPP into Matrix rooms and DMs
    - Libpurple
      - Bridges anything that libpurple can speak into Matrix! (like Bitlbee, but for Matrix)

Other new stuff!

- Ultra-Low-Bandwidth Matrix
  - Targeting 100bps links(!)
  - Uses fan-out routing to minimize global bandwidth
  - Should get added to the Matrix spec.
  - Come to RTC Dev Room (H.1309) tomorrow at 10:40 to hear more.
- Dendrite
  - Our Golang homeserver implementation is ticking away
  - We hope to (eventually) get it up to parity with Synapse now 1.0 is done.

What's After 1.0?

- Performance and Resource Usage:
  - Incremental State Resolution (massive time, CPU & RAM savings)
  - Chunked Room Representation (reduce fragmentation/extremities!)
  - State compression (massive disk space savings)
  - Sharding Synapse per room
- Features:
  - Retention schemes
  - Admin interface
  - Reworking Communities
  - Reactions and Editable messages
  - Extensible Profiles
  - Decentralised Accounts
  - Removing MXIDs
  - Finish RiotX (and write RiotX/iOS)
  - Threading
  - P2P Matrix & better routing
  - Decentralised reputation
  - Decentralised identity

[matrix]

We need help!!

[matrix]

DON'T USE PROPRIETARY SERVICES FOR YOUR CHAT.

- Run a server, or use a provider like modular.im
- Build bridges and bots to your services!
- Don't reinvent the wheel, use Matrix!
- Follow @matrixdotorg or @matrix@mastodon.matrix.org and spread the word!

Thank you!

@matthew:matrix.org
matthew@matrix.org
https://matrix.org
@matrixdotorg