

[matrix]

Matrix for MMI

matthew@matrix.org | travis@matrix.org

@matthew:matrix.org | @travis:t2l.io

**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's mission:
To build the real-time
communication layer of the
open Web.**

Matrix as a Framework

I-D.ralston-mimi-matrix-framework

Matrix's capabilities

- Matrix is already used for interoperable chat!
- Rooms are replicated* across all participating servers
- ACLs, history retention, etc are all defined by the room version
- Events carry information between users in a room
- Extensible support for current & future chat features
- Supports end-to-end encryption, including (D)MLS
- HTTP+JSON APIs for ease of use, but can be (nearly) anything

*uses a DAG normally, but Linearized Matrix fixes that...

Eventual consistency

- Allows any server in the room to go offline and receive events later
- Offline servers only affect users on that server
- Uses a Directed Acyclic Graph (DAG) to store events, like Git
- “Gaps” in the DAG are healed when servers come back online
- Plenty of rules to govern how the DAG accepts events
- Not exactly the easiest thing to implement in DMA’s short timescales

What MIMI looks like on Matrix

[matrix]

MIMI	Matrix
Messaging framework ACLs, permissions, basis for interoperability	Room versions Contains the algorithms needed to run the room
Messaging Content/Format What is ultimately encrypted or sent over the wire	Extensible Events Uses “content blocks” to reuse schemas
Transport Between backend servers	HTTP+JSON primarily Could be (nearly) anything, though
User discovery Including resolving a phone number to a user ID	Identity servers We'd rather replace these with something else
MLS for E2EE	DMLS for E2EE, or optionally Olm/Megolm

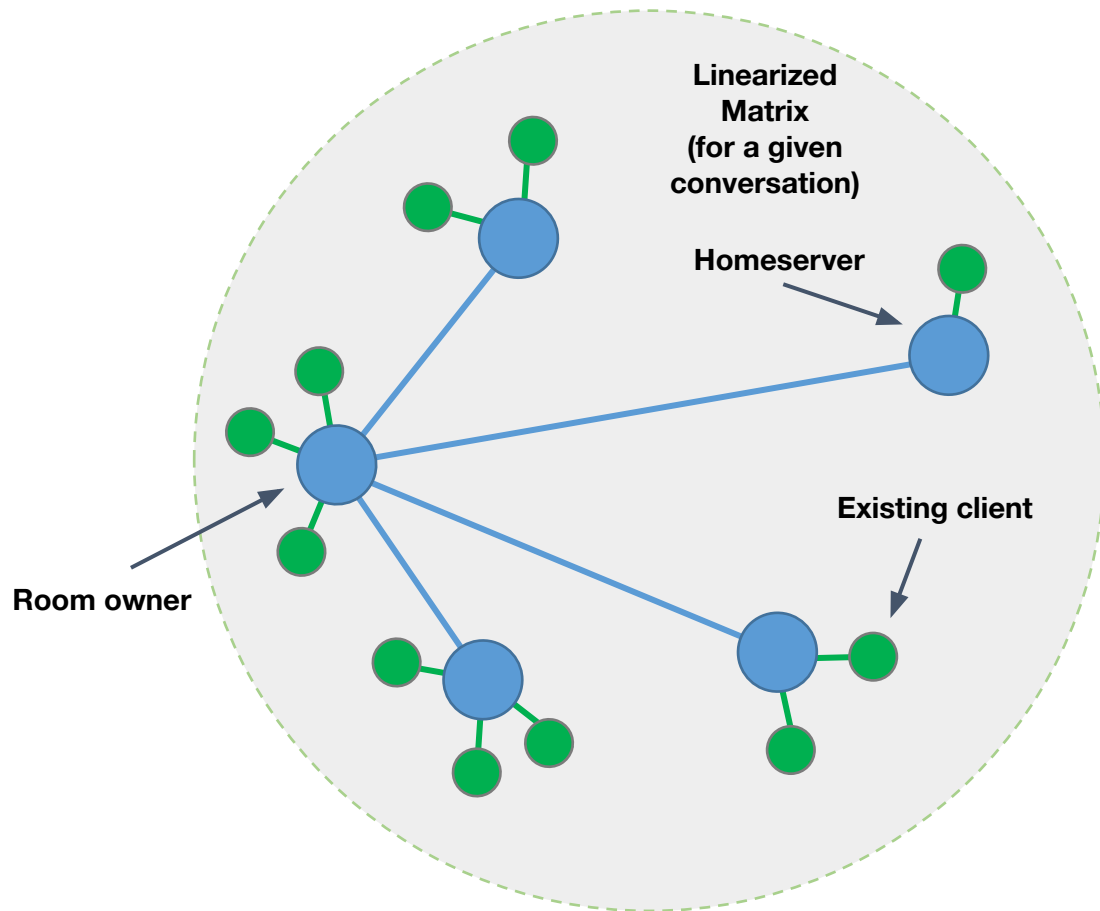
**What if we didn't have to
use a DAG though?**

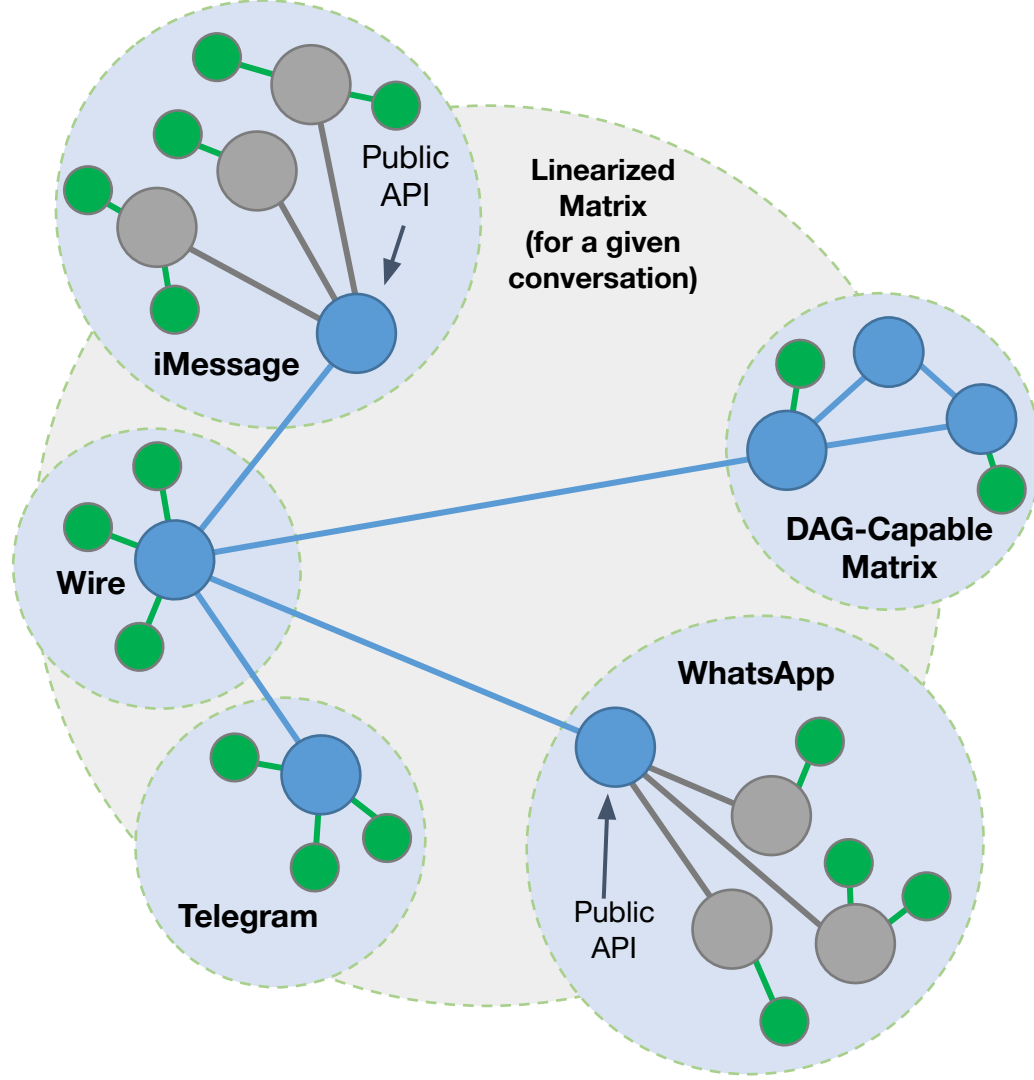
Linearized Matrix

I-D.ralston-mimi-linearized-matrix

Linearized Matrix

- Partially centralizes rooms around an owner for a given conversation
- Still uses Matrix's access control semantics and rules
- Still uses Matrix's event format for interoperability
- Represents rooms as a single array rather than a DAG
- Significantly easier to implement and reason about
- Put it over whatever transport you like
- Doesn't shut the door on decentralisation; can interop with normal DAG-capable Matrix.





[matrix]

Thank you!

@matthew:matrix.org | @travis:t2l.io
matthew@matrix.org | travis@matrix.org
<https://matrix.org>