Defragmenting the Internet for fun and non-profit
First, let's consider our origins...
What was the original promise of the Internet?
An interoperable platform for sharing data.
What is the reality today?
User data is trapped in vendors.
Services are getting more and more fragmented...
The user experience is broken...
For example:

What does it mean to "call someone via WebRTC"?
This message contains an invitation to an event.

From Tsahi Levent-Levi
Subject Invitation: Kranky Geek @ Thu Apr 9, 2015 3:30pm – 3:55pm (tsahi.leventlevi@gmail.com)
To Me <matthew@matrix.org>

Tsahi Levent-Levi has invited you to Kranky Geek

Title: Kranky Geek
Location: https://beta.talky.io/tsahil
Organiser: Tsahi Levent-Levi <tsahi.leventlevi@gmail.com>
Description: View your event at https://www.google.com/calendar/event?action=VIEW&amp;eid=Z2c2b3BmbGxtNzUxcTRkb25qZDBvZGEyZTggbWF0dGlld0BtYXRyaXgub3Jn&amp;tok=MjYjdHNhaGkubGV2ZW50bGV2aUBnbWFpbC5jb204NTdINGE2NjBhNmM2ZGU1NGUzZDAntcz=Asia/Jerusalem&amp;hl=en.
Attendees: Tsahi Levent-Levi <tsahi.leventlevi@gmail.com>

1 attachment: invite.ics 1.1 kB
| Title:          | Kranky Geek                          |
| Location:      | https://beta.talky.io/tsahil          |
| Organiser:     | Tsahi Levent-Levi                    |
Why should I have to use Tsahi's app?
Why should I trust Tsahi's app?
Why should the data for this call get trapped in Tsahi's app?
How do I know how to contact Tsahi in future?
And similar situations exist for IoT, IM, and any type of data!
The Internet is cracking up!
Imagine a world where data is owned by the user and securely shared across services
What would it take?
A federated open standard like Email and the Web...
...but for data exchange.
A federated open standard for data exchange...

- Backed by a strong community
- Endorsed by various industries
- As pragmatic as SMTP
- As open as HTTP
Digging a bit further:
Data owned by the user

A federated open standard for data exchange

Data securely shared across services

- Distributed architecture
- Federated architecture
- Strong security via E2E crypto
- Ability to be used over various transport protocols
- Agnostic to the type of transmitted data
- Easy to implement (pragmatic, open)
- Easy to deploy
- Strong identity system
Enter Matrix:
Aiming to provide the missing federated open standard for data exchange
What is Matrix?
Non-profit Open Standard / Open Source (Apache Licensed) project
Provide an open global network for securely exchanging and storing real-time data.
Could be WebRTC Signaling
Or Group Messaging
Or IoT Data
A simple HTTP API layer on top of the existing web, that provides a consistent simple way to publish and subscribe to data in real-time.
All data is securely decentralised using a blockchain-style system - not stored in any single place, under any single owner: no SPOF/SPOC.
Everything is group communication, and all communication has persistent history.
Matrix realises WebRTC's full potential as a universal media layer for the Internet
We have a choice: create new islands or create interoperable services.
Sometimes islands make sense
But sometimes interoperability and federation makes sense too!
Where are we at?
September 2014 - Disrupt SF
Launched Alpha (v0.2)

- Initial draft of v1 Spec and APIs
- Apache-Licensed Python reference implementation (Synapse)

- Basic Angular web client implementation for group chat and VoIP.
- Client SDKs for Perl
Nov. 2014 - WebRTC World San Jose
Entered Beta (v0.5)

- Feature complete for core group chat & VoIP features
- Full TURN support for WebRTC
- Initial iOS and Android "Console" applications
- Strong crypto (encrypting and signing) all federated traffic
- Client SDKs for iOS (CocoaPods), Android, Python, JS, Angular
Since November:

- Launched Application Services API: how you bridge other networks and functionality into Matrix.
- Published mobile apps in the app stores
- Reusable UI SDKs for iOS "MatrixKit" and Android
- Added Decentralised content repository (mxc:// URLs)
- Added Push notifications
- Added performance metrics
- Initial IoT examples
- V1 spec improvements
- Designed some V2 APIs to fix the uglier bits in V1
- Lots and lots of bug fixes
Most interestingly:

- Explosion of 3rd party clients, application services and even servers implementations.
- Clients
  - MatrixKit UI
  - Weechat
  - React
- Application Services
  - SIP: Clearwater (telco v. internet example)
  - IRC: gatewaying
- Servers
  - Pallium and jSynapse
Clearwater SIP GW Example

Project Clearwater (IMS Core) <-> Gateway (IBCF) <-> Local Matrix Home Server

- SIP
- SIP/WebSockets Gateway
- SIP/Websockets
- jsSIP (SIP client)

WebRTC

- Federation
- Remote Matrix Home Server
- Matrix
- Matrix Web UI
Today we're proud to announce a major stable beta approaching 1.0 release (v0.9)
Version 0.9

- **Lots** of performance work
- Supporting Postgres as well as SQLite
- All new configuration system
- Lots of long tail fixes (you can now reset your password!)

- Experimental support for mobile support for VoIP through OpenWebRTC
What's next?

- End-to-end encryption
- Reusable web UI components and improving the web client
- Multi-way VoIP
- Lots more Application Services
- Landing V2 APIs
- Use 3rd party IDs by default
- Yet more performance work
- Spec polishing
- New server implementations!
Why should you use Matrix?
The more people who build on Matrix, the more useful it is.
This should be a no-brainer!
It's Free!

(both commercially and philosophically)
Free reusable UI components and WebRTC support on Mobile!
Free client applications for you to use as a starting point for your own apps!
Free server implementations to power your messaging and VoIP
Don't waste time inventing your own HTTP signalling protocol when we've done it for you!
An ever-larger growing community of users for you to on-board into your app!
You can own and control your own data!
Pick interoperability as a differentiator for your app!
How to use & help Matrix?
Pick a client and try it now!

http://matrix.org/blog/try-matrix-now
Run a server!

http://matrix.org/blog/try-matrix-now
Build an application service to bridge in your existing apps

http://matrix.org/blog/try-matrix-now
Consider native Matrix support for new apps.
Or just help us spread the word (@matrixdotorg)!
Thank you!

matthew@matrix.org

http://matrix.org

@matrixdotorg