Experimenting with Matrix federation over Yggdrasil

Bachelor Semester Project

Timothée Floure
Responsible / Prof. Bryan Ford
Supervisor / Cristina Basescu

EPFL / DEDIS

2020-01-13

Big Picture

- Matrix: modern, federated (instant) messaging system.
- Yggdrasil: experimental P2P overlay network, providing E2EE and compact routing (ish).

Aim of this project: have **matrix homeservers exchanging over the Yggdrasil network**.

Matrix + Yggdrasil

- Project discussed with Matthew Hodgson from Matrix.org.
- Neil Alexander from Yggdrasil was interested to support this project.
- Personal interest in Matrix and P2P systems.

Matrix 1/2

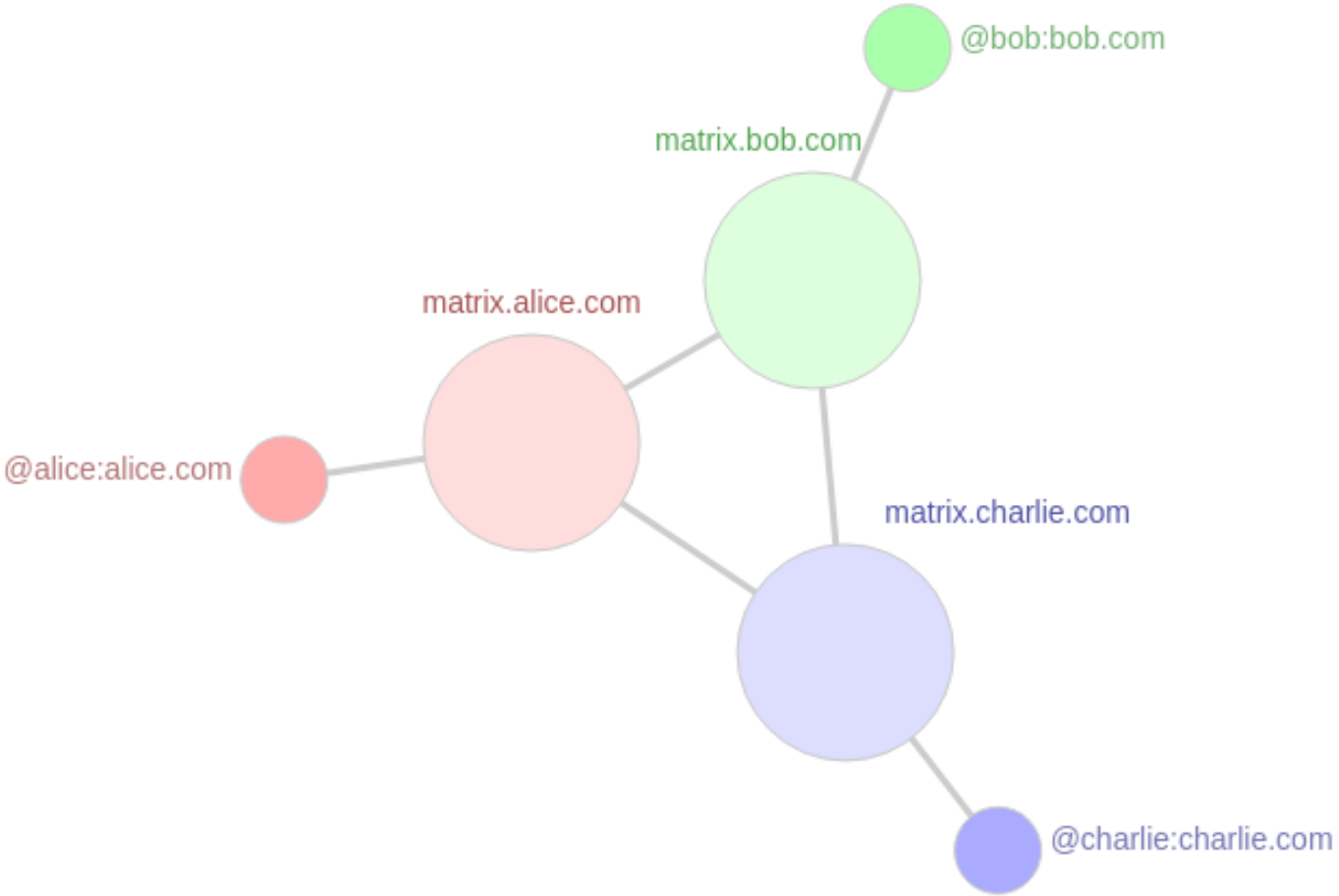


Figure 1: Three Matrix homeservers, each with one client connected.

Matrix 2/2

- Modern feature: media support, message history, end-to-end encryption, VoIP bridging to other IM services.
- 1.0 milestone reached during the first half of 2019.
- Communications done over HTTP(S).
- A few rough edges, but active community and going in the right direction!

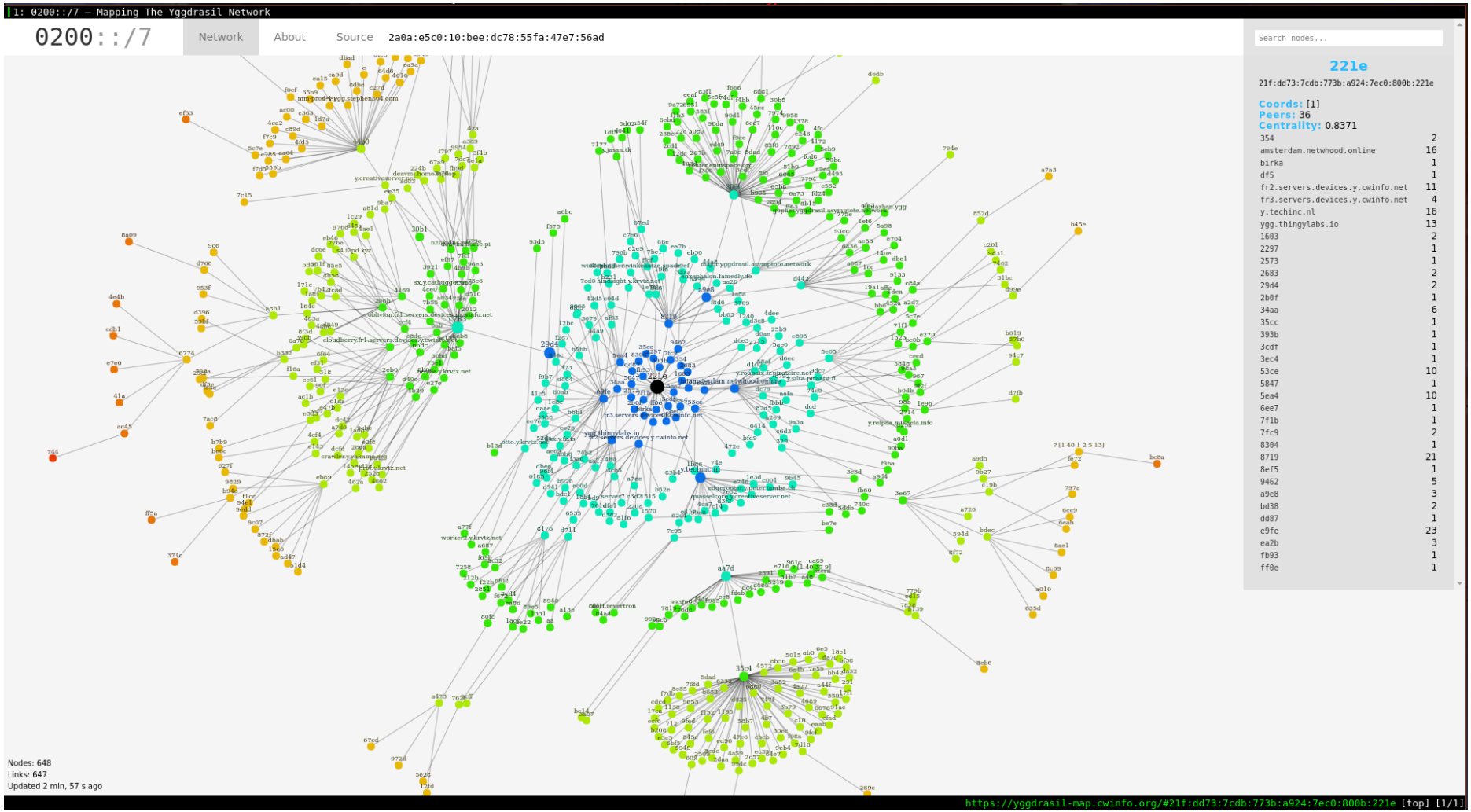
Yggdrasil 1/3

- **Experimental** P2P, end-to-end encrypted, self-arranging overlay network.
- Routing inspired from Robert Kleinberg's *Geographic Routing Using Hyperbolic Space*.
- No automatic peering management or naming system.

Yggdrasil 2/3

- Can be embedded into applications using Go library (August 2019).
- Draft specification published (mid-fall 2019).

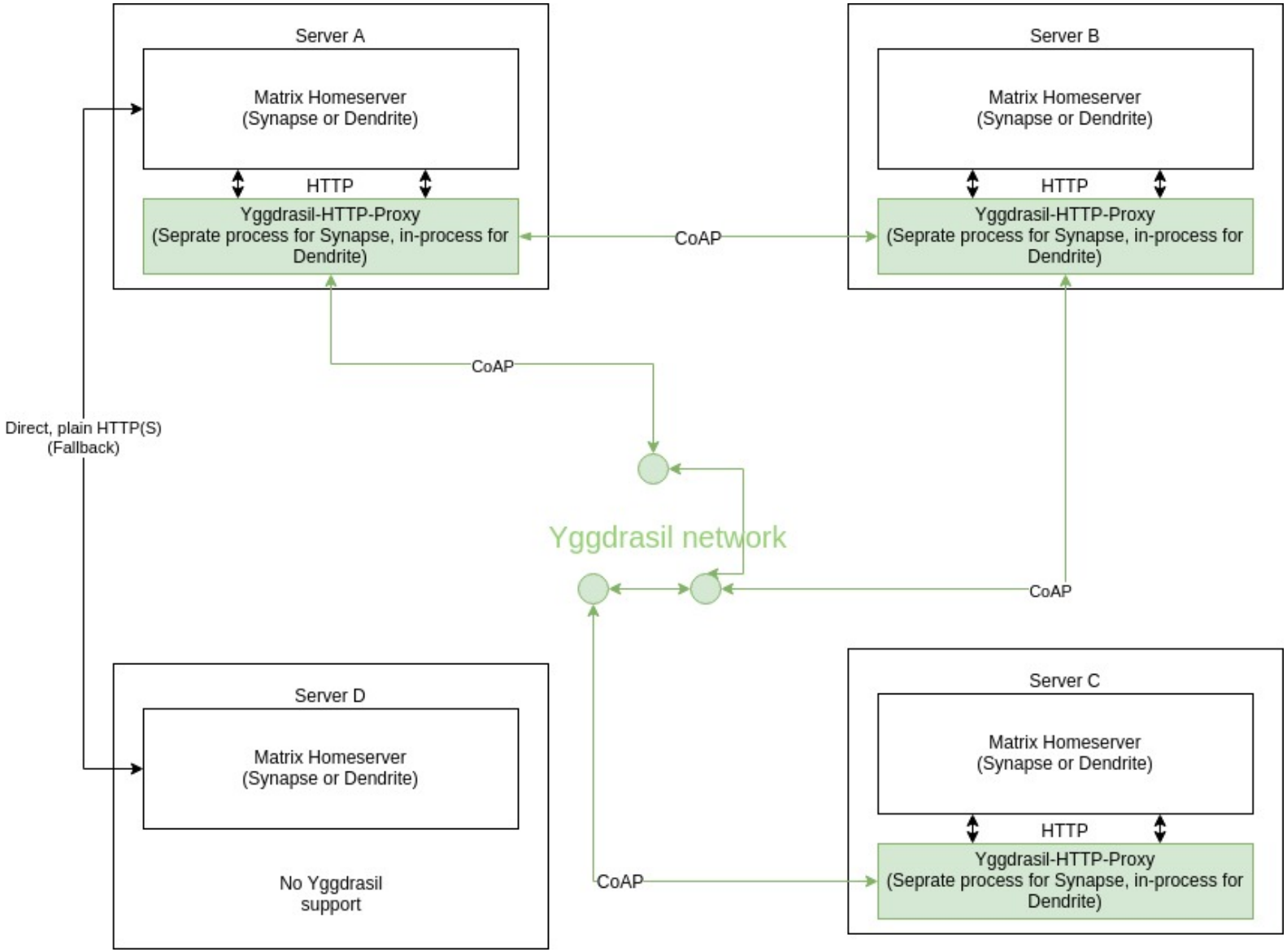
Yggdrasil 3/3



Matrix + Yggdrasil

- Integration with Matrix homeserver.
- Yggdrasil address resolution.
- HTTP over Yggdrasil.
- Practical Yggdrasil peering.

How does it look?



Integration with Matrix homeserver

- Initially wanted to use Dendrite, next-gen homeserver written in Go.
- Dendrite federation was not working properly yet.
- Switched to use Synapse, the reference (python) homeserver implementation.
- Direct integration was not possible anymore (no python Yggdrasil library).
- => HTTP-over-Yggdrasil-and-Back proxy

Yggdrasil address resolution

- NodeID (sha512sum of encryption key).
- Falling back to standard DNS insfrastructure.

HTTP over Yggdrasil

- Initially wanted to leverage Go's net/http.
- Realized that Yggdrasil exposes connections more akin to UDP than TCP.
- CoAP (**CO**nstrained **A**pplication **P**rotocol) as a translation layer.

CoAP as a translation layer

- Defined over UDP by RFC7252.
- Build with low-bandwith, unreliable links in mind.
- REST-like!
- Previous low-bandwith experiment by matrix.org.
- go-ocf/go-coap library.

Practical Yggdrasil peering

- No automatic peering update / bootstrapping.
- Peers discovered from actual traffic (DNS).
- Periodic routine updating peers based on latency/usage/stability.
  - Work-in-progress.

Output

- go-ocf/go-coap fork adding Yggdrasil support.
- matrix-org/synapse fork adding federation proxying + yggdrasil NodeID resolution.
- matrix-yggdrasil-http-proxy.

It works!™

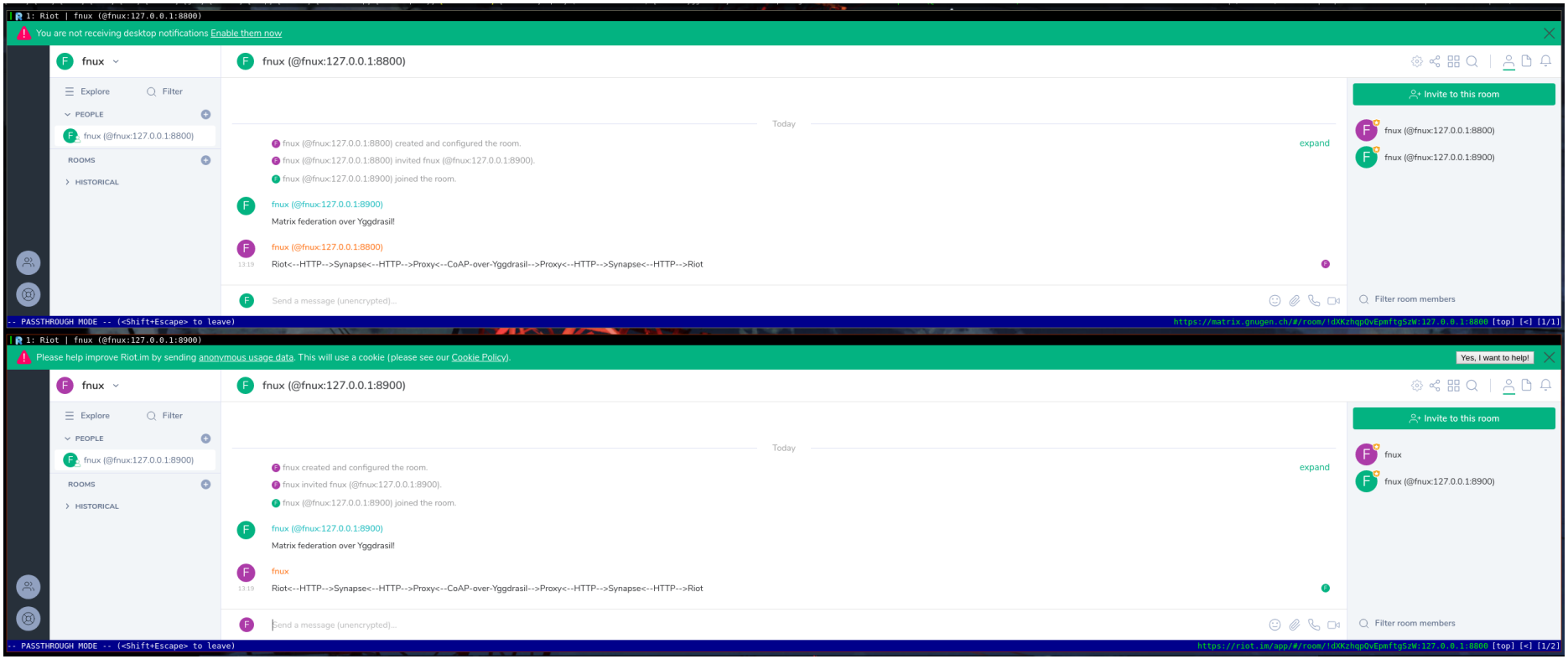


Figure 4: A working proof-of-concept

Future work

- Compression!
- Peer selection.
- In-browser homeserver & Dendrite integration.
- Homeserver discovery and name resolution.
- Extended testing and real-world usage.

=> Will likely be discussed at FOSDEM.

Thanks

- Cristina Basescu from DEDIS.
- Neil Alexander and Arceliar from Yggdrasil.
- Matthew Hodgson from the Matrix.org Foundation.

Wrapping up: Matrix over Yggdrasil

- Matrix: federated IM system.
  - Synapse reference (python) homeserver, Dendrite next-gen (Go) homeserver.
- Yggdrasil: E2EE, self-arranging P2P overlay network. Routing inspired from Robert Kleinberg's work.
- CoAP: low-bandwith REST-like protocol defined over UDP.
- matrix-yggdrasil-http-proxy: HTTP to CoAP over Yggdrasil and back.