



sip:element@matrix.org

October 16 2025

Unifying Telephony and Modern Communication



Who am I ?



Laurent Meiller

@lmeiller · Member since January 3, 2017

Overview

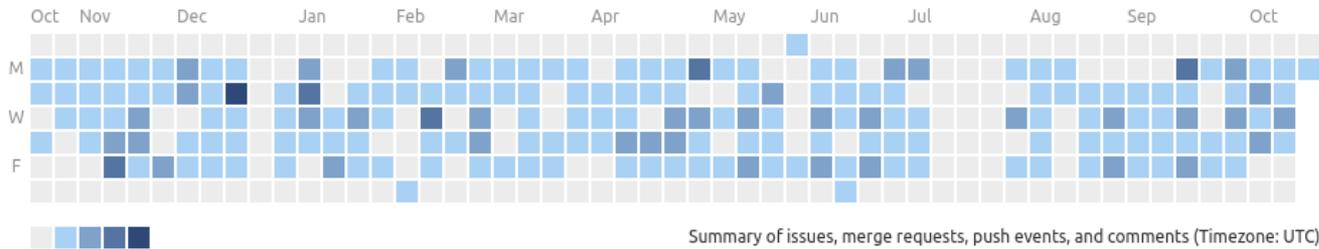
Activity

Groups

Contributed projects

Personal projects

Snippets



Who are we ?

XiVO is a French **editor** and **integrator** of an **open-source** solution in the IP **Telephony** market.



Key points of XiVO solution :

Full IP telephony (calls, transfers, conferencing built on Asterisk)
Native Interoperability (API first)
A robust, proven platform used in demanding environments
(public sector, healthcare, defense)



First, some key figures in France

In 2024, MS Teams was adopted by **132,000** companies and public administrations.

For comparison, at the same date, INSEE recorded **159,000** SMEs (> 10 people).

But... what's the point with Telephony ?

- Loss of PSTN connectivity breaks inbound/outbound calling with externals.
- Users must juggle between chat and legacy phone systems.

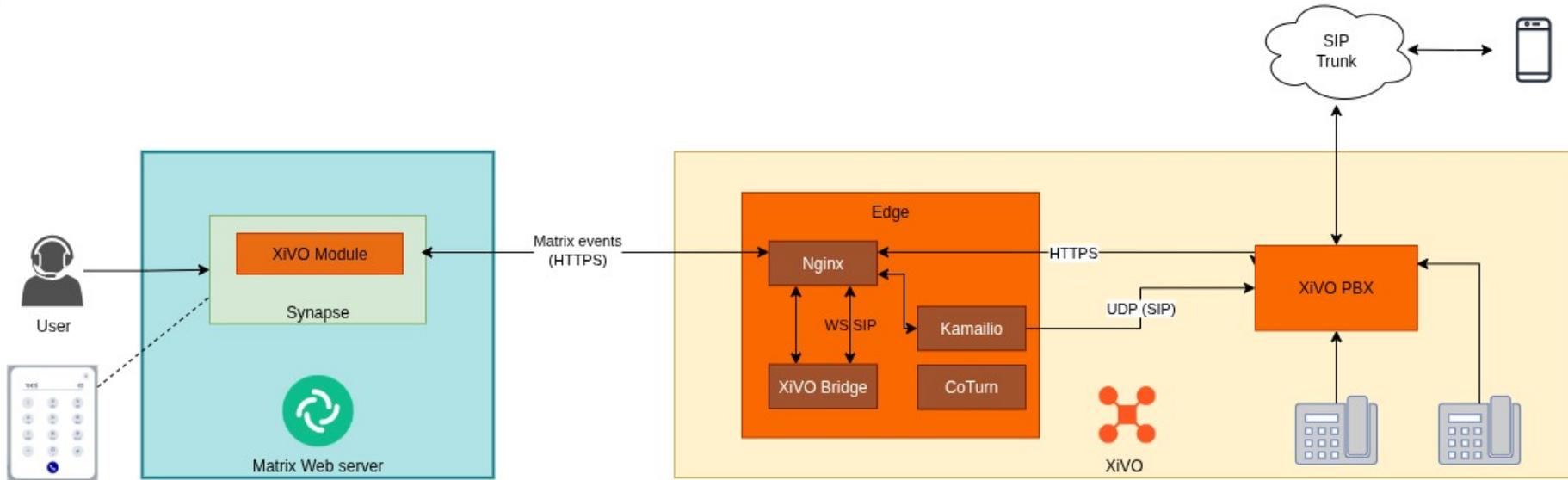
Our Goal ?

Unify existing VoIP and legacy telephony into a single **Element** environment to take care of change management.

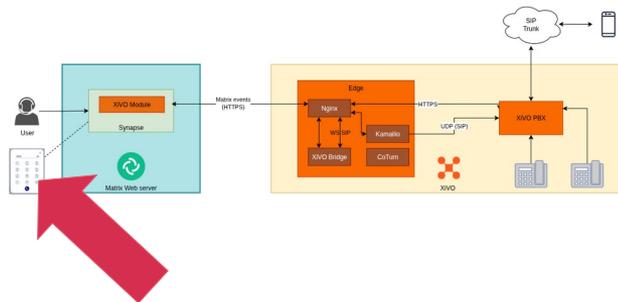


Demo

Architecture Overview



Let's rewind a bit... Why not use a "simple" widget integration ?



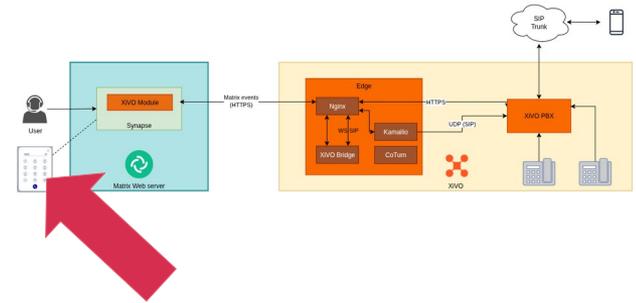
Widget Pros

- Embedding interactive UI (calls, history, configuration...)
- Launching calls within a room (à la Element Call)

And Cons

- Not persistent, usually tied to a single room
- Limited mobile and Element X support
- Not suitable for incoming calls

So, how to move forward then ?



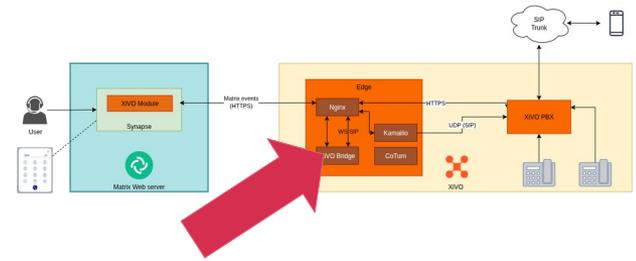
```
dbkr Clarify existing thirdparty spec fa50766 · 5 years ago History
```

Preview Code Blame 96 lines (76 loc) · 4.02 KB

MSC2845: Thirdparty Lookup API for Telephone Numbers

Matrix clients may support VoIP and bridges can exist that accept a Matrix VoIP call and terminate it on the PSTN, but currently no provision is made for a user to enter a telephone number in order to call that number.

Application Service (AS) is the key !



```
51 # Optional
52 # The protocols the application service supports.
53 protocols:
54   - m.protocol.pstn
55   # - im.vector.protocol.sip_virtual
56   # - im.vector.protocol.sip_native
57
```

Just need to translate **Matrix** events to **SIP** messages, easy right... right ?

SIP ?

SIP manages **signaling**, meaning:

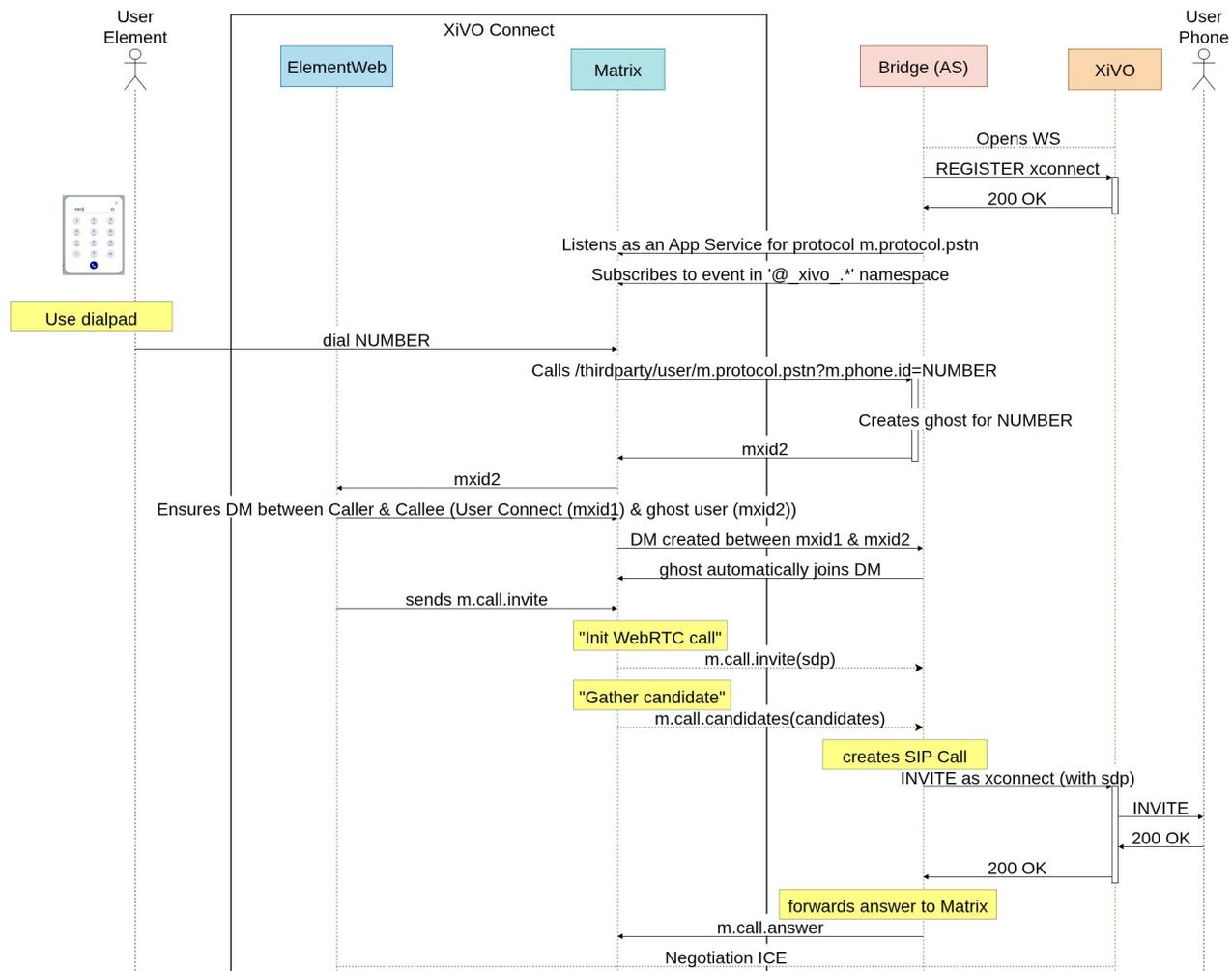
- Who is calling whom
- Where the user can be reached
- What media (audio, video, codecs) to use
- How to end or transfer the session

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.com
s=-
c=IN IP4 192.0.2.101
t=0 0
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

```
Alice (UAC)      Proxy/Server      Bob (UAS)
|--- INVITE ----->|
|<-- 100 TRYING -----|
|<-- 180 RINGING -----|
|<-- 200 OK -----|
|--- ACK ----->|
| [Échange RTP : audio/vidéo] |
|--- BYE ----->|
|<-- 200 OK -----|
```

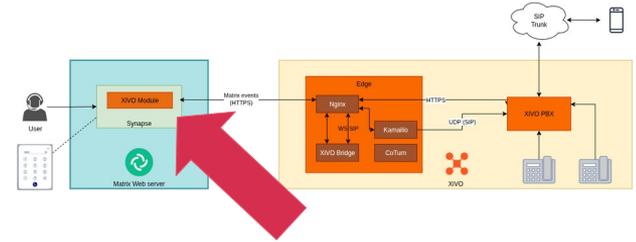
SIP vs WebRTC ?

Aspect	SIP - XiVO	WebRTC - Matrix LegacyCallHandler
Purpose	Legacy telephony interco	1:1 communication in the browser
Signaling	Defined by the protocol (INVITE, ACK, BYE, etc.)	m.call.* (offer/answer)
Media transport	RTP (optionally SRTP)	SRTP (encryption mandatory)
NAT traversal	Optional	Built-in STUN/TURN/ICE
Security	Optional (TLS, SRTP)	Mandatory (DTLS-SRTP, HTTPS)
Interoperability	Compatible with other PBXs, phones, trunks	Compatible with browsers and native apps
Integration complexity	High (requires SIP server, proxy, etc.)	Low (built-in JS APIs)

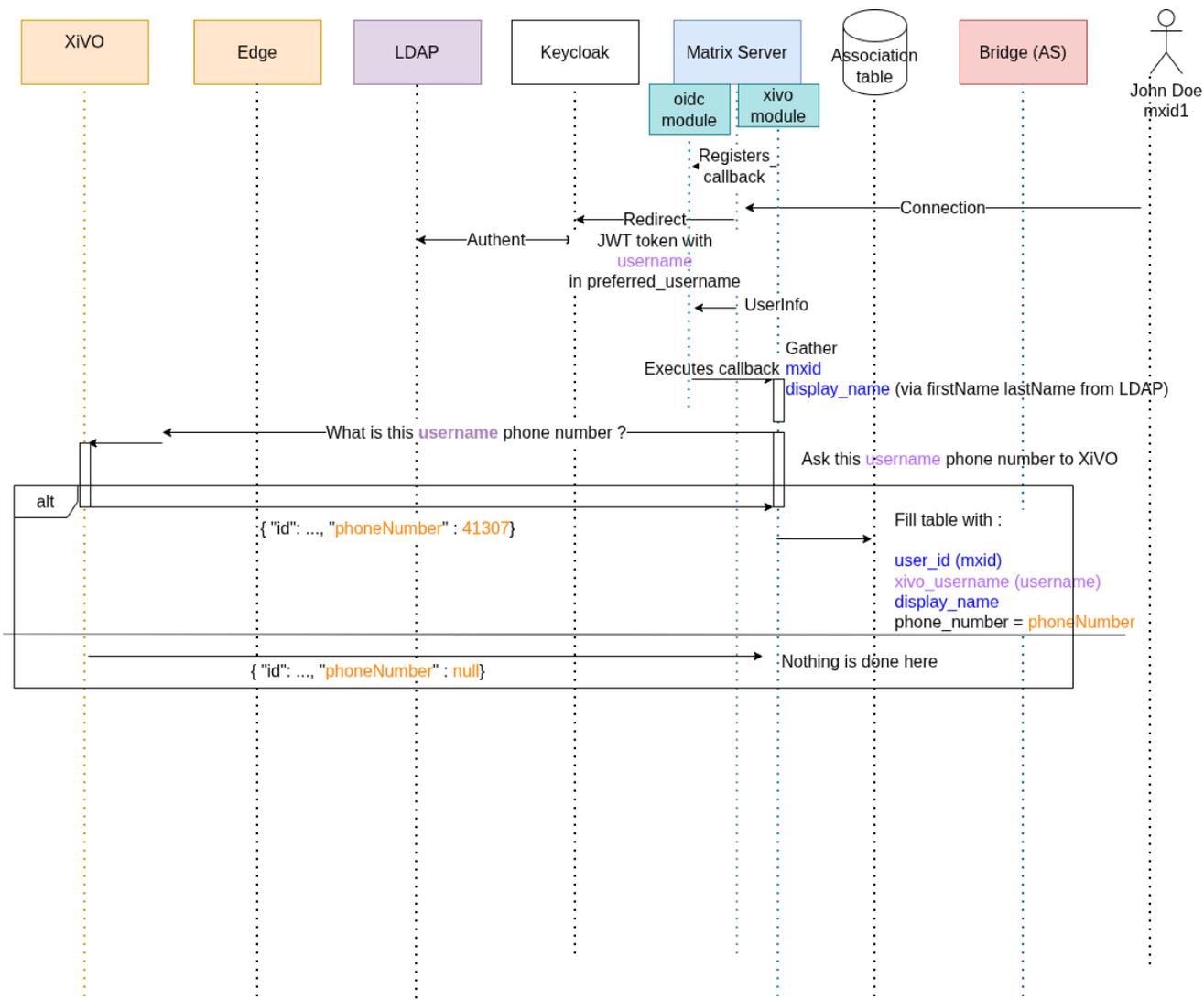


So what ?

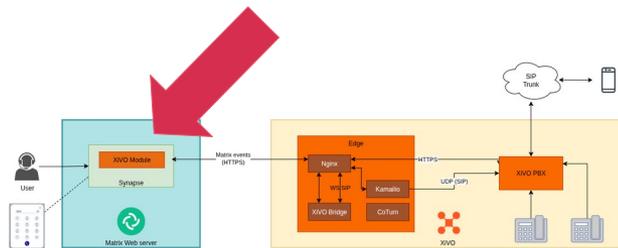
Cool, we managed outgoing call, but how to make incalls ?



We need a mapping between
XiVO users and **Mxid** !!!

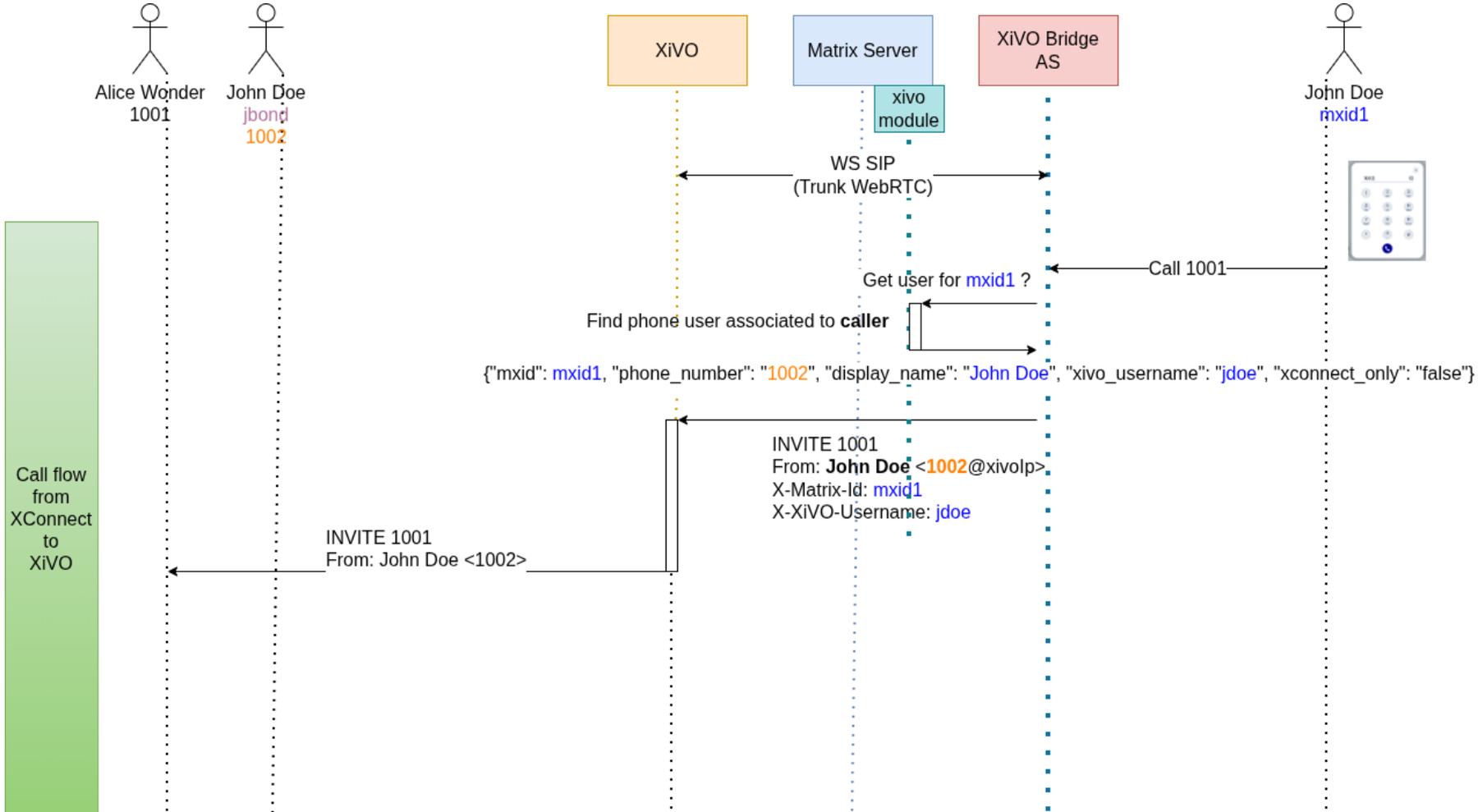


Let's do more !

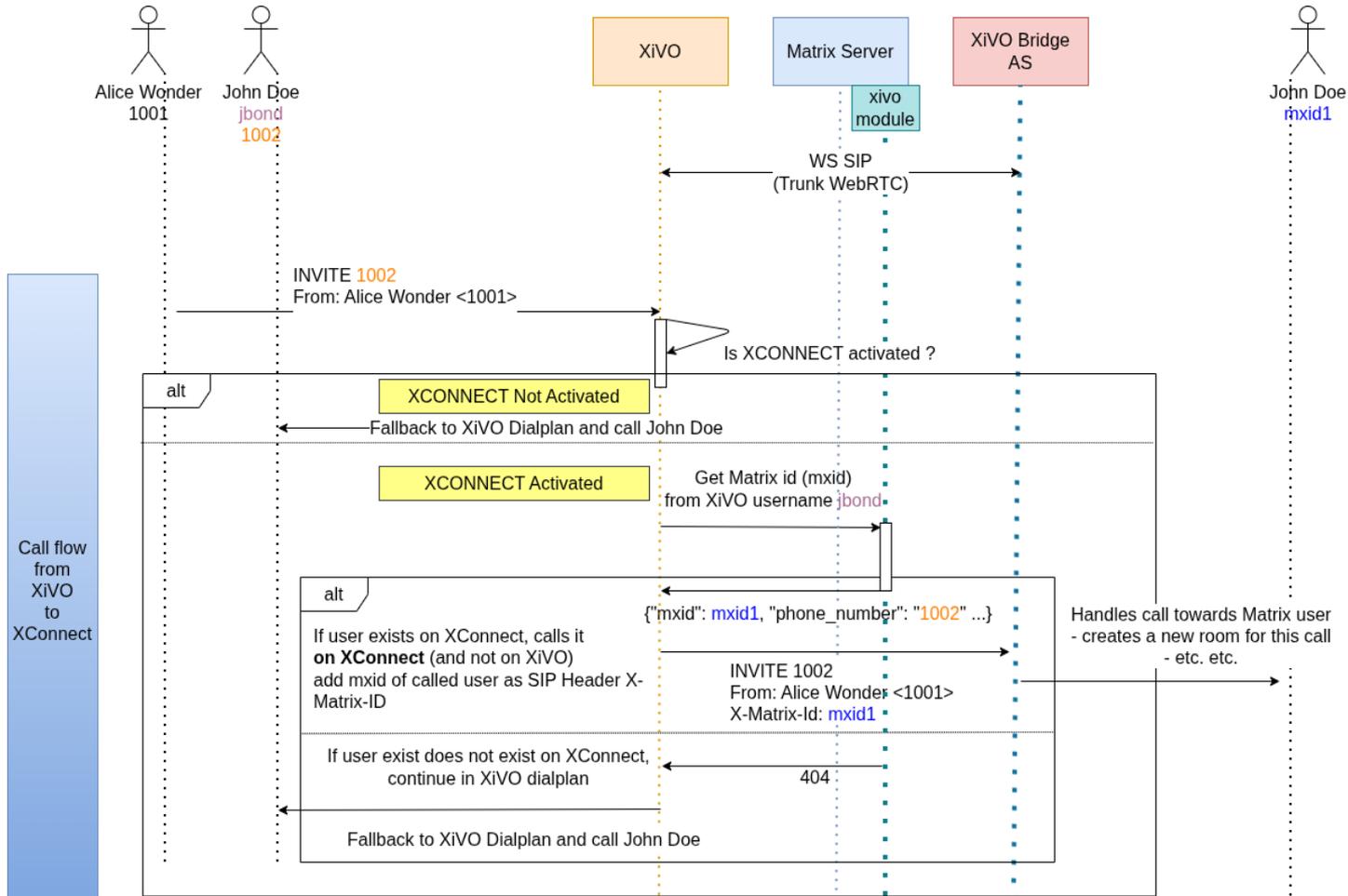


```
if self.is_a_telephony_room_invite_event(event):  
    await self.tweak_telephony_room_permissions(event)  
    await self.add_telephony_room_to_telephony_space(event.room_id)  
elif await self.is_a_telephony_room_call_invite_event(event):  
    await self.log_call(event)  
elif await self.is_a_telephony_space_or_history_room_leave_event(event):  
    await self.invite_to_room_by_service_account(event.sender, event.room_id)
```

Wrap up for outgoing call !



Wrap up for incoming call !



What about future ?

- Create widget for line configuration (DND, forwards, telephony presence...)
- Decouple XiVO synapse module and replace it by a bot (Tchap initiative)
- Make it compatible with MatrixRTC ?
- Heavily load test to the limits !!

Any ideas ?

Thank you !

<https://gitlab.com/avencall/xivo-connect>



XIVO