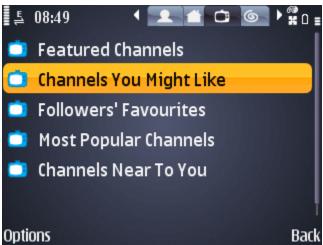
The API is dead. Long live the protocol.











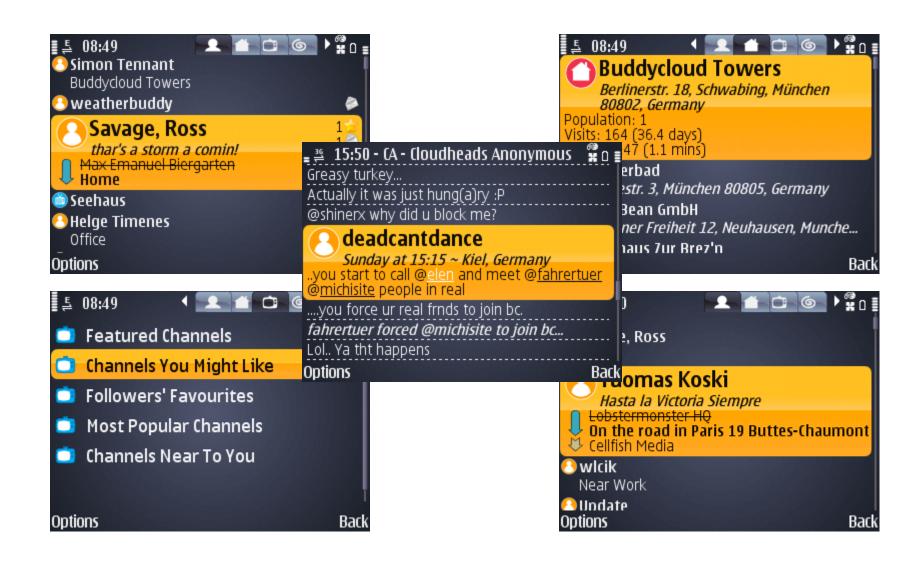






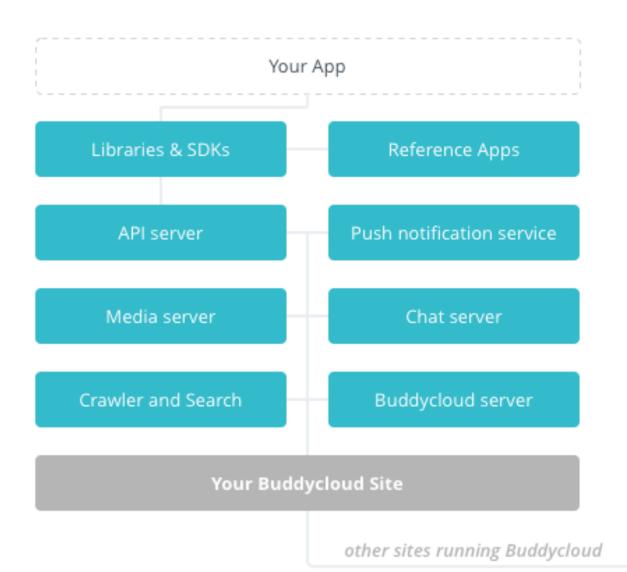






Example 8. Recent items reply

```
<iq to="juliet@capulet.lit/bc-app" from="buddycloud.capulet.lit" type="result" id="recent-items:1">
 <pubsub xmlns="http://jabber.org/protocol/pubsub">
   <items node="/user/romeo@montague.lit/posts">
       <item id="tag:channels.capulet.lit,/users/juliet@montague.lit/posts,16584564008760001">
            ...atom payload...
       </item>
       <item id="tag:channels.capulet.lit,/users/juliet@montague.lit/posts,10070506450002">
            ...atom payload...
       </item>
   </items>
   <items node="/user/juliet@capulet.lit/posts">
       <item id="tag:channels.capulet.lit,/users/juliet@capulet.lit/posts,3252545492409">
            ...atom payload...
       </item>
   </items>
   <items node="/user/romeo@montague.lit/posts">
       <item id="tag:channels.capulet.lit,/users/juliet@montague.lit/posts,106876700003">
            ...atom payload...
       </item>
   </items>
 </pubsub>
</ig>
```



We need to add messaging to...



Developers don't know who to trust or have a clear methodology.

Messaging is complex and developers spend a lot of time reinventing the wheel.

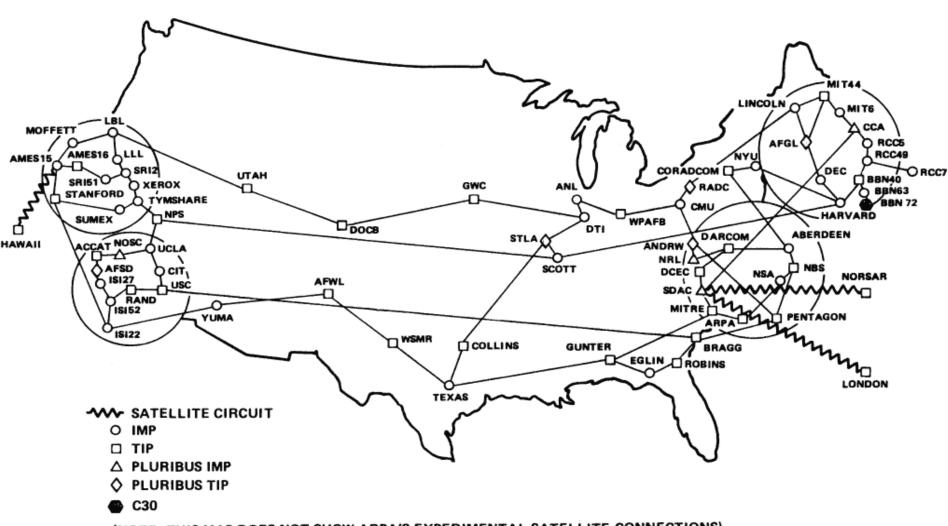
Running and scaling messaging... where to start?





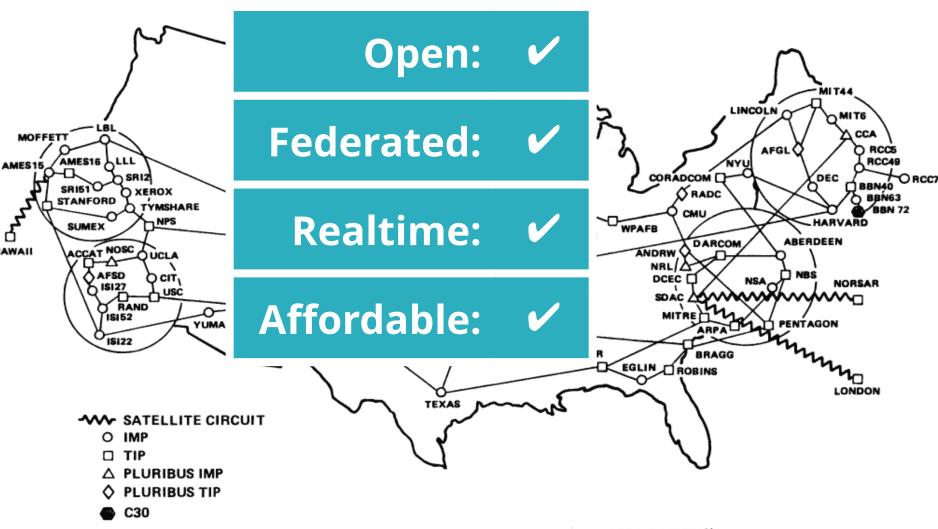


ARPANET GEOGRAPHIC MAP, OCTOBER 1980



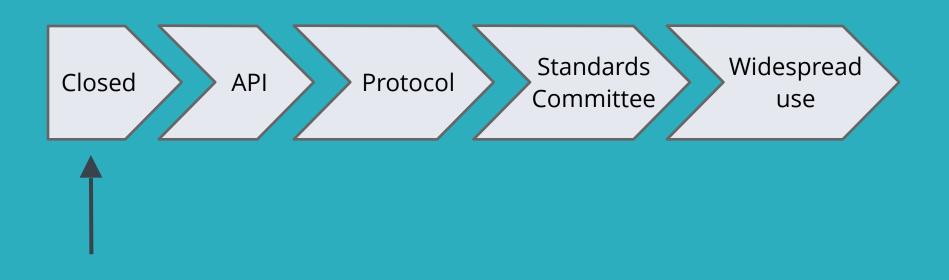
(NOTE: THIS MAP DOES NOT SHOW ARPA'S EXPERIMENTAL SATELLITE CONNECTIONS)
NAMES SHOWN ARE IMP NAMES, NOT (NECESSARILY) HOST NAMES

ARPANET GEOGRAPHIC MAP, OCTOBER 1980

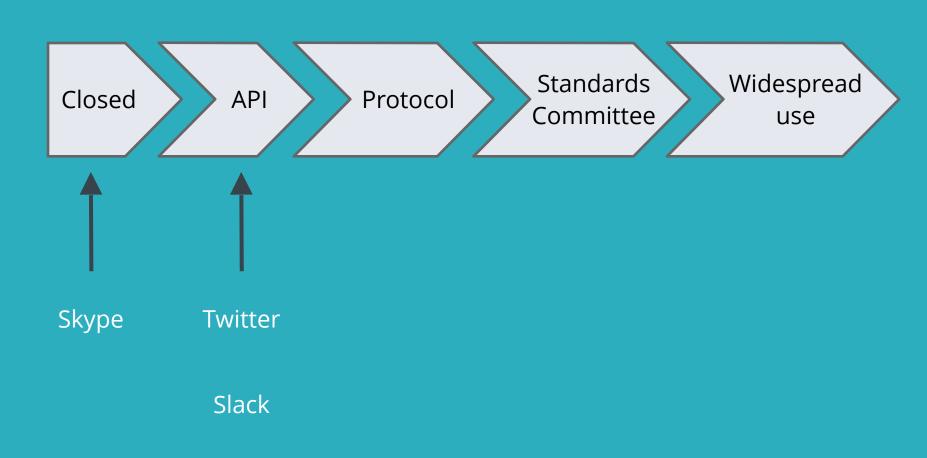


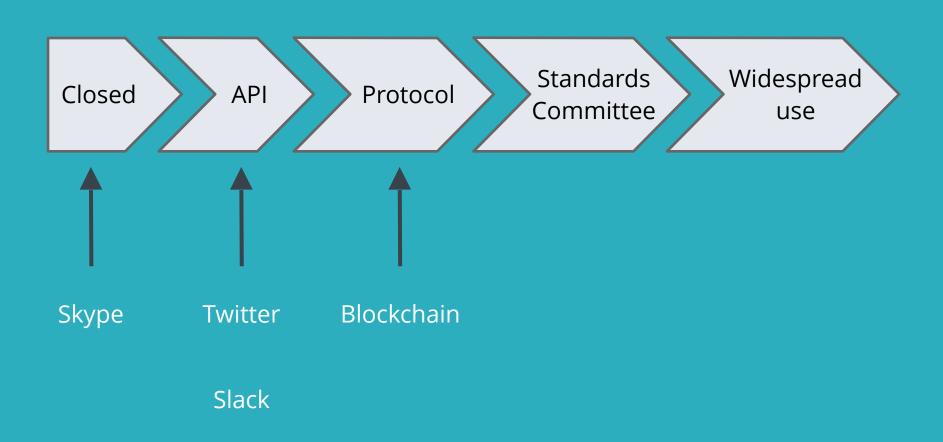
(NOTE: THIS MAP DOES NOT SHOW ARPA'S EXPERIMENTAL SATELLITE CONNECTIONS)
NAMES SHOWN ARE IMP NAMES, NOT (NECESSARILY) HOST NAMES

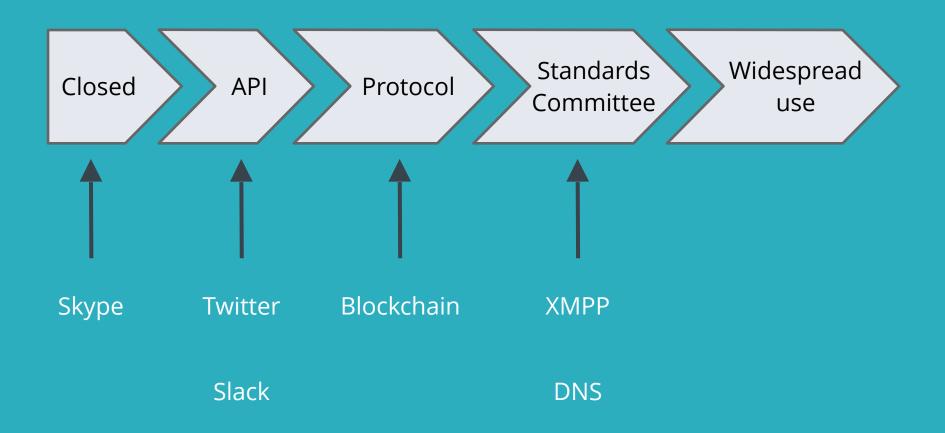


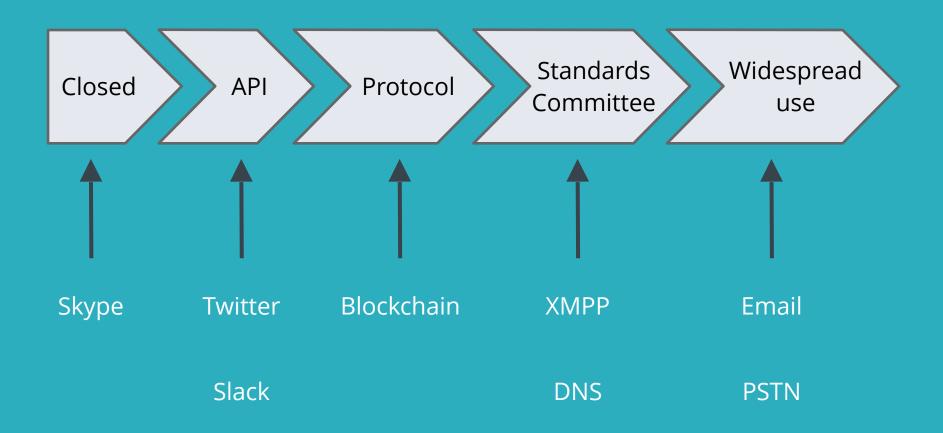


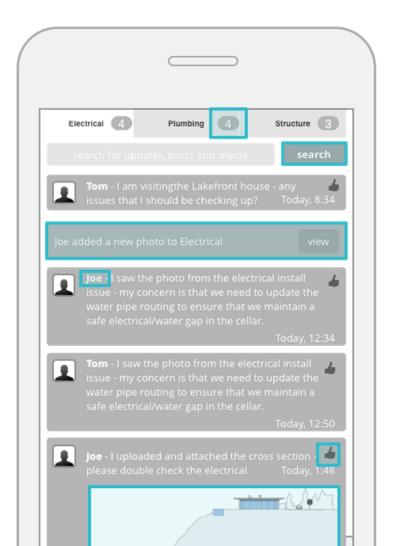
Skype









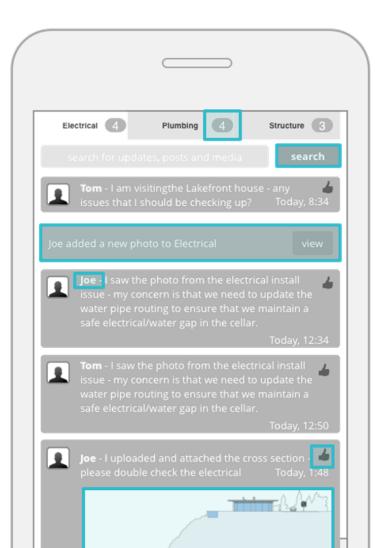


Backend Services

User Management

Push

Search

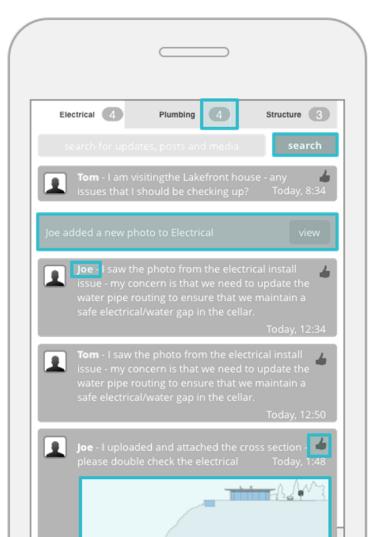


Backend Services

User Management

Push

Search







User Management

Push

Search

Hosting

Scaling

Security

Portable data







User Management

Push

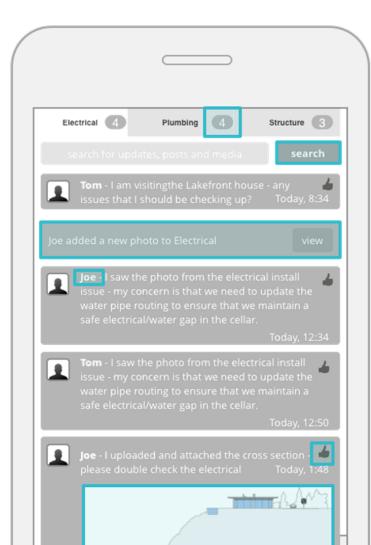
Search

Hosting

Scaling

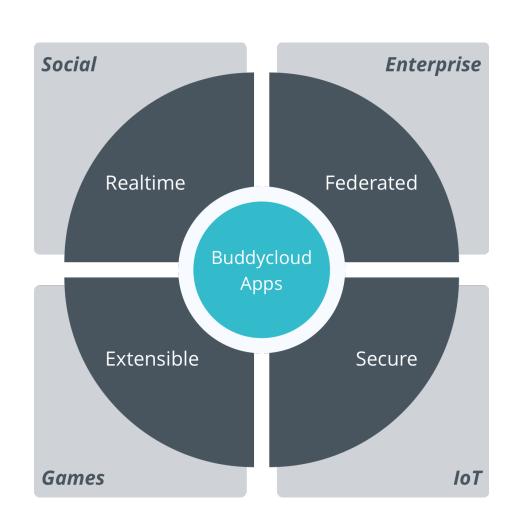
Security

Portable data



Client SDK Group messaging Social activity stream File sharing













Evolution of in-app messaging Erector Set → Fit for purpose

2013 2015







everything you need to know about XMPP



Used by







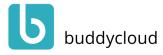




Real Time



- Real Time
- Identity



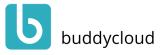
- Real Time
- Identity
- Security



- Real Time
- Identity
- Security
- Scalability



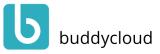
- Real Time
- Identity
- Security
- Scalability
- Discovery



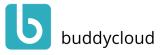
- Real Time
- Identity
- Security
- Scalability
- Discovery
- Interoperability



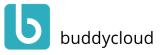
- Real Time
- Identity
- Security
- Scalability
- Discovery
- Interoperability
- Authentication



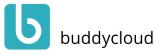
- Real Time
- Identity
- Security
- Scalability
- Discovery
- Interoperability
- Authentication
- Authorization



- Real Time
- Identity
- Security
- Scalability
- Discovery
- Interoperability
- Authentication
- Authorization
- Provisioning



- Real Time
- Identity
- Security
- Scalability
- Discovery
- Interoperability
- Authentication
- Authorization
- Provisioning
- Presence



romeo@montague.lit

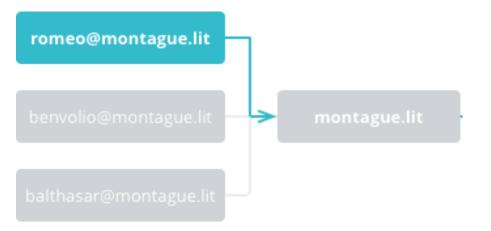
benvolio@montague.lit

balthasar@montague.lit

juliet@capulet.lit

nurse@capulet.lit

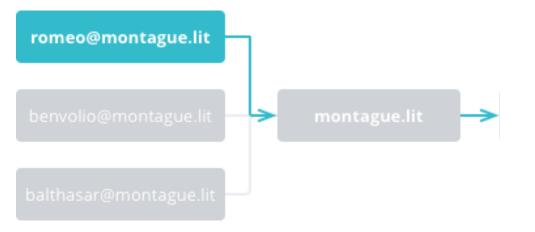
tybalt@capulet.lit



juliet@capulet.lit

nurse@capulet.lit

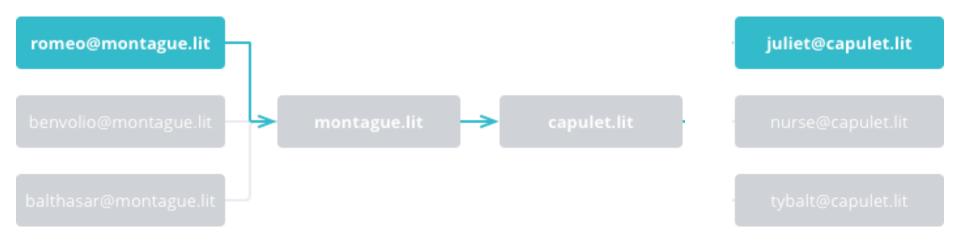
tybalt@capulet.lit

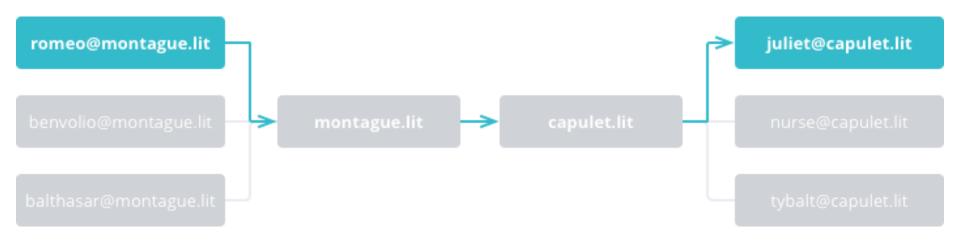


juliet@capulet.lit

nurse@capulet.lit

tybalt@capulet.lit





```
<message from='romeo@montague.lit'
to='juliet@capulet.lit'
type='chat'>
<body>We have had a most delightful evening, a most
excellent ball.</body>
</message>
```

```
C: <?xml version='1.0'?>
   <stream:stream
       to='example.com'
       xmlns='jabber:client'
       xmlns:stream='http://etherx.jabber.org/streams'
       version='1.0'>
S: <?xml version='1.0'?>
   <stream:stream
       from='example.com'
       id='someid'
       xmlns='jabber:client'
       xmlns:stream='http://etherx.jabber.org/streams'
       version='1.0'>
     encryption, authentication, and resource binding ...
     <message from='juliet@example.com'
C:
              to='romeo@example.net'
              xml:lang='en'>
       <body>Art thou not Romeo, and a Montague?</body>
C:
C:
     </message>
     <message from='romeo@example.net'</pre>
S:
              to='juliet@example.com'
              xml:lang='en'>
       <body>Neither, fair saint, if either thee dislike.</body>
S:
     </message>
S:
C: </stream:stream>
S: </stream:stream>
```



XML?



XML? Where's my REST API?



XML?
Where's my REST API?
Mobile and Push?



XML?
Where's my REST API?
Mobile and Push?
I want to send a file?



XML?
Where's my REST API?
Mobile and Push?
I want to send a file?
Just chat?

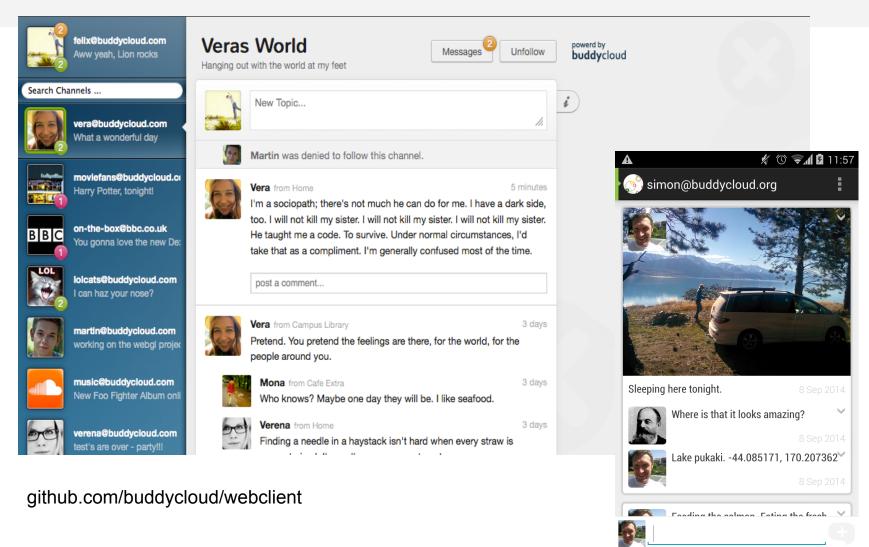


Everything you need to know about Buddycloud

github.com/buddycloud

Buddycloud extends XMPP

Demo apps

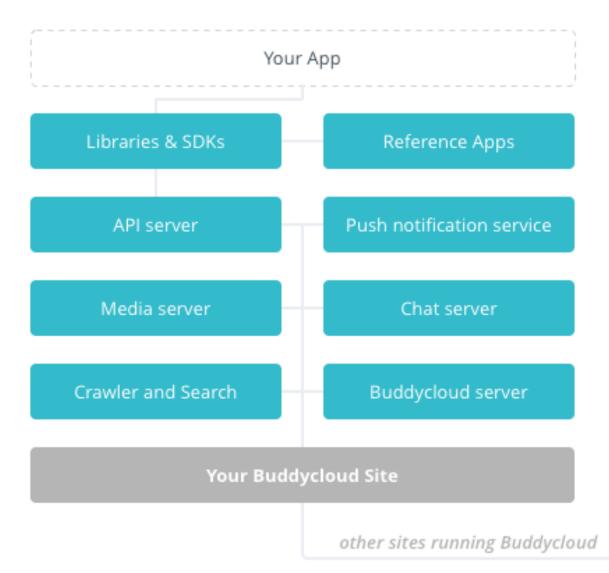


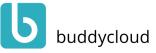
buddycloud

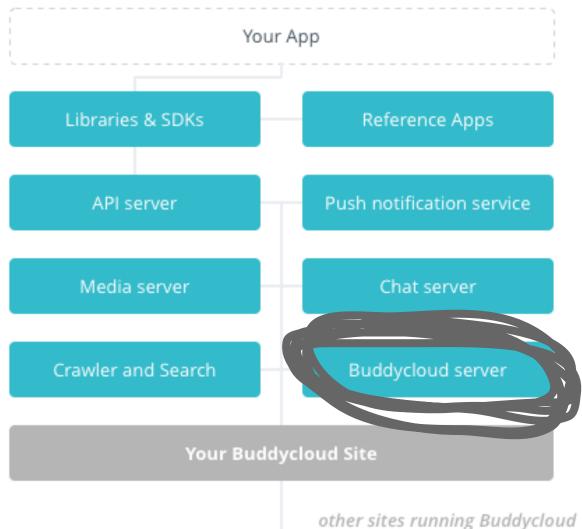
A small diversion into microservices

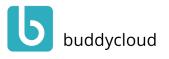
A small diversion into microservices

In computing, microservices is a software architecture style, in which complex applications are composed of small, independent processes communicating with each other using language-agnostic APIs. These services are small, highly decoupled and focus on doing a small task.



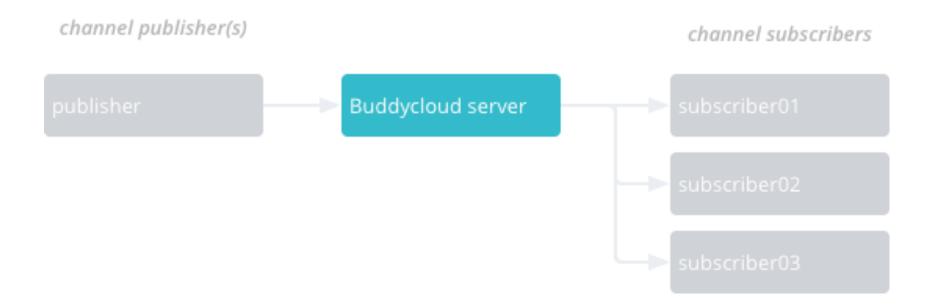






other sites running Buddycloud

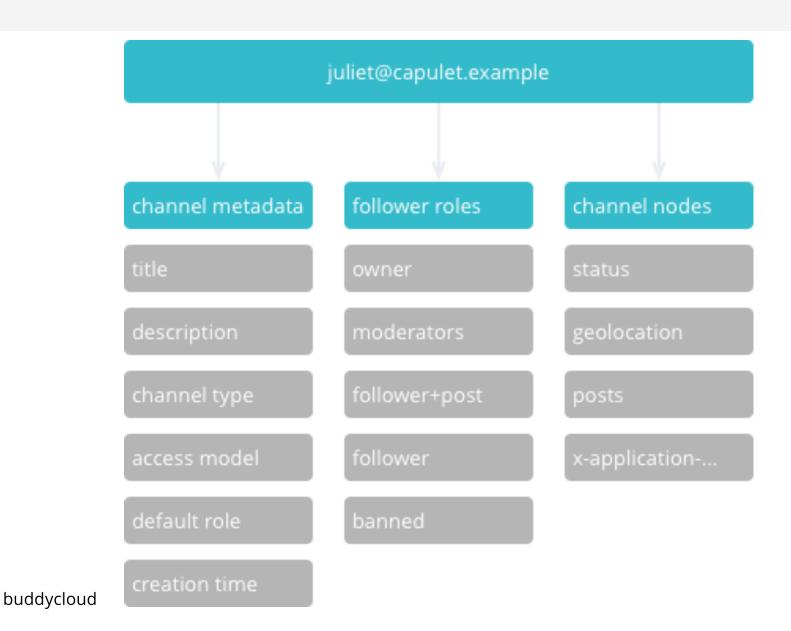
Everything is Publish-Subscribe





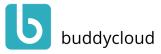
/posts (microblog) /geolocation /<your application>







Problems



Problems

slow discovery and retrieval



Problems

- slow discovery and retrieval
- networks aren't always online



Problems

- slow discovery and retrieval
- networks aren't always online



Problems

- slow discovery and retrieval
- networks aren't always online

Solution: The Inbox approach

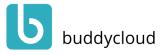


Problems

- slow discovery and retrieval
- networks aren't always online

Solution: The Inbox approach

improves time to glass



Problems

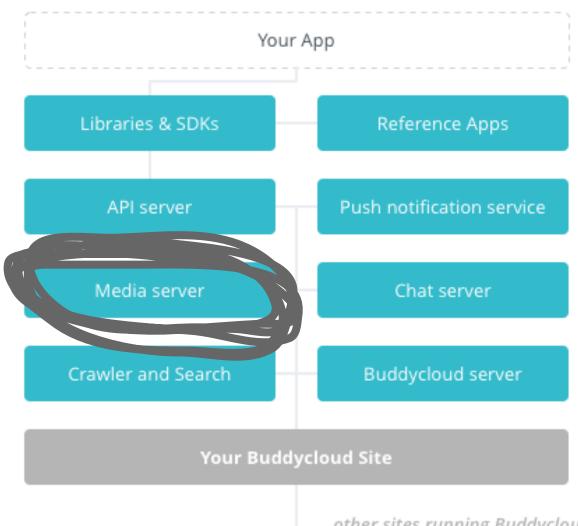
- slow discovery and retrieval
- networks aren't always online

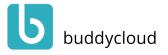
Solution: The Inbox approach

- improves time to glass
- catches up on missed posts when servers come back online



Looking at the media server

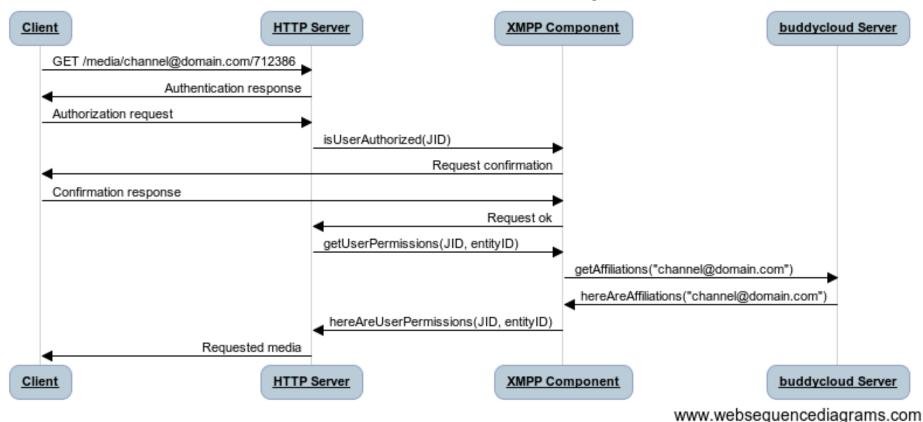




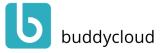
other sites running Buddycloud

Requesting media

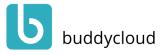
Successful Authentication Sequence







API server



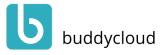
- API server
- Push server



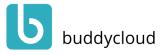
- API server
- Push server
- Friend finder



- API server
- Push server
- Friend finder
- Taste engine



- API server
- Push server
- Friend finder
- Taste engine
- Search



Where next?

Make it easy to add messaging...

Leverage open standards to avoid lock-in and save time.

Flow: Slack-clone for customers with "behind the firewall" needs.

Standardise more Buddycloud services.