XmppTalk

Philippe Back
@philippeback

Pharo
What is XMPP?

eXtensible Messenging and Presence Protocol

The most secure messaging standard

XMPP is the open standard for messaging and presence

XMPP powers emerging technologies like IoT, WebRTC, and social.

It’s a living standard. Engineers actively extend and improve it.

No one owns XMPP. It’s free and open for everyone since 1999.

Millions use XMPP software daily to connect to people and services.
Uses of Xmpp

XMPP is the Internet Standard for Instant Messaging, Real Time Messaging, and Multi-User Chat. It provides high functionality and good security capabilities. XMPP is being increasingly adopted by NATO and National organizations, as the technology of choice for Instant Messaging.

Scenarios

Two sample scenarios are considered that show ways in which XMPP can be used, and requirements that need to be considered given the mix of communication methods likely to be available during a deployment.
So, I wanted an XMPP client in Pharo for the Slack

Since then we moved to Discord and there is no XMPP there.

Anyway...
One needs a server

http://files.pharo.org/media/logo/logo-flat.png
Unpack and run OpenFire (Needs some Java)

```
philippe@ubuntu:/opt/openfire$ ls
bin documentation LICENSE.html README.html
changelog.html embedded-db logs resources
conf lib plugins
philippe@ubuntu:/opt/openfire$ cd bin
philippe@ubuntu:/opt/openfire/bin$ ll
```

```
total 36
-rwrxr-x 1 philippeback philippeback 13388 May 6 2014 openfire*
-rwrxr-x 1 philippeback philippeback 5034 May 6 2014 openfirectl*
philippe@ubuntu:/opt/openfire/bin$ ./openfire start
Starting openfire
philippe@ubuntu:/opt/openfire/bin$
```
One needs a non Pharo client
One does not want to recode the low level

libstrophe

An XMPP library for C

libstrophe is a minimal XMPP library written in C. It has almost no external dependencies, only an XML parsing library (expat or libxml are both supported). It is designed for both POSIX and Windows systems.
Build 32 bit library for Pharo 32-bit

Use a container (LXC)

Can do it with Docker as well. No need for docker with LXC.
One needs to bind to Pharo: UnifiedFF UFFI Binding for Libstrophe
Interesting...

Typedefs

Constants

Callbacks
One needs a Pharo based client: XmppClient
One has to map callbacks
In order to implement XEPs
One has to see what is going on

Modeled after ZnLogEvent thing but will move to Beacon.
Raw stuff can also be logged !console) and one can thus see the data stream

SESSION xmlns="urn:ietf:params:xml:ns:xmpp-session"/>features>
conn DEBUG SENT: <iq id="_xmpp_bind1" type="set">bind xmlns="urn:ietf:params:xml:ns:xmpp-bind"<resource>PharoIDE</resource></bind></iq>
xmpp DEBUG RECV: <iq id="_xmpp_bind1" to="vmfractal/f1b4d050" type="result">bind xmlns="urn:ietf:params:xml:ns:xmpp-bind"<jid>pharo@vmfractal/PharoIDE</jid></bind></iq>
xmpp DEBUG Bind successful.
conn DEBUG SENT: <iq id="_xmpp_session1" type="set">session xmlns="urn:ietf:params:xml:ns:xmpp-bind" xmlns:xmpp-session="/"></iq>
xmpp DEBUG RECV: <iq id="_xmpp_session1" to="pharo@vmfractal/PharoIDE" type="result"/>
xmpp DEBUG Session establishment successful.
conn DEBUG SENT: <presence/>
xmpp DEBUG RECV: <presence to="pharo@vmfractal/PharoIDE" from="pharo@vmfractal/PharoIDE"/>
conn DEBUG SENT: </stream:stream>
2017-05-18T11:42:10.296709+02:00 -- Disconnected
event DEBUG Stopping event loop.
2017-05-18T11:42:10.298184+02:00 -- Stop for context done
conn DEBUG Can’t reset connected object.
2017-05-18T11:42:10.298564+02:00 -- Connection released
2017-05-18T11:42:10.29864+02:00 -- Context Freed
2017-05-18T11:42:10.298694+02:00 -- Shutdown done
It is annoying to use test cases for interactive exploration

Using RunCase approach
One needs a GUI, one day

Using Spec

Leveraging the announcements

Target is have a one to one chat and a MUC (Multi User Chat)
Conclusion

Pharo proved to be suitable for this

Very easy to implement callbacks

Promising for new XEPs

Stable

Easy to debug
Horizons

Have a couple XMPP servers in the cloud

Communicate right form the IDE