

Microblogging workflow

The number of accounts I have on useful networks, social or otherwise, has been on an increasing trend-line for a while. In the beginning, things were simple, but it gets messy pretty quick. One feature which these networks share in some form or another are *status notices*

Starting with the 'chat status' in the Instant Messaging world (like: **I'm busy** or **Away from computer** etc.) to the slightly more verbose microblogging notices which describe what the user is doing or finds interesting in about 140 characters or less.

So, if you have a handful of accounts and you want to update your status or give a notice of something. What do you do? Obviously, you'd want to avoid going to all those sites and doing them one by one.

Posting is one thing...

ping.fm has part of the solution. Their idea is to create one point to send status updates and micro blog notices and they will take care of delivering to all the other networks you are a part of. A huge improvement over posting the same content on all the networks separately.

However, it's only part of the solution. Granted, in the domain of microblogging alone the way to post should be readily available (otherwise the moment is gone) and the content is meant to be short and restricted, so redistribution is likely to succeed to many places as the requirements will be low. Perfect when the only origin of your content is microblog-like. Mine is not, however.

The way authoring works is by choosing a publishing means and place based on the content of the message you are trying to get across to a certain audience. For microblogging, typically short and 'for everyone who is interested' this can be anything capable of sending out 140 characters to the nearest more connected hub with a supported API. After that, a [ping.fm][ping] like service can do the rest. For a blog—entry (like this), not suitable as microblog content as such, there needs to happens something additional. The content that should go to the social networks is not the actual content of the blog—entry, but more a notification that it happened, posted in the nicest possible way, and in realtime, as the current times demand.

So, with that, is the problem then reduced to the following two-step?

1. if the content is microblog like: use ping.fm and be done with it;
2. If the content is not suitable for microblogging, use a specific tool and place and make sure a notification goes out to ping.fm with a back reference.

Authoring needs more

Sure, if no-one ever talked back, this would be all that is needed. However, the apparent idea behind social networks is that people **do** talk back. If the 'just playing with this thing'—stage is over and people start talking back, you're back into the old situation where you visit all the networks separately to manage the replies to your notices. In fact, it's worse. I could, if forced, live with publishing in many places if it meant getting all replies in one (visible) place. In reality all the replies (in whatever form) are scattered all over the place.

Now what? Aggregate in RSS-like form by creating extensions on all the networks to make it one feed? Bring yet another service into the mix? Use an omnipotent client on every platform you work on? Just accept that you cant communicate with everyone in the same way and that it just takes time?

We need a pong.fm as the counterpart of ping.fm!

Can we do better?

In the domain of instant messaging I have more or less solved the problem by installing our own jabber server with gateways to icq, aim, gtalk, other jabber servers etc. This gets me to communicate with everyone, regardless of their choice of IM network, whereas I can use my favourite IM client on all platforms without hassle everywhere. (Quite a bit of hassle on the server though). The interface is the same for all IM networks, I can initiate and reply in the same way and I don't need to know anything else; the server is connected to all the networks (obviously I need to register to those networks once on the server, so the server knows where to bring the message) Everything looks like a jabber JID to me in the IM-domain. I want something similar for microblogging!

From the top of my head, such a system should satisfy at least the following:

- be fully distributed, so others can do 'the same' and we can federate and scaling is trivial;
- be open and standardised, so we all know what 'the same' means;
- be able to integrate with the main microblogging initiatives out there in such a way that it is transparent for all communicating parties (multi-directional);

(I'm ignoring any functional specifications for the moment, just what it should be at an architectural level)

A couple of existing systems popped into my mind while jotting down the above points. First, google wave, mostly because of its distributed real-time multi-user

authoring, their solution for the storage issues involved and a couple of other innovations which seem to fit the above (like basing it on XMPP, making it open and having extension-options on both server and client side).

Another system which I thought of was identi.ca (or rather laconi.ca which is the code-base that runs identi.ca). This actually **is** a microblogging system which already supports cross-posting and federation. Their openness is attractive as well as their initiatives to standardise the microblogging protocol for interoperability.

It's still early days for both of these systems however.

Will I be able to run a laconi.ca/wave install on our own servers, "gateway" it to our XMPP server and every major other μ -blogging platform out there and just register with my own server(s) and will it feel as if I participate in **one** community as a whole, where I can communicate with everyone, regardless of their choice in whatever the latest popular community is?

I think the answer is, and will always be: **"Not yet, but we're close"**. It is just another iteration of the 'Unified Messaging' paradigm. We're getting closer each run, but messaging gets redefined over time, making us chase the end of the rainbow again, once we've (partially) fulfilled an earlier goal.