# The Rights Ratchet Model

## Bait & Switch meets Boiling Frogs

**Simon Phipps,** Meshed Insights Ltd

*Upstream Conference, June 7, 2021*

# Journey Of An Open Source Pioneer

2004    SugarCRM Open Source Project founded, Mozilla+attribution licensing
        Company is VC funded, open core business model
        25,000 Downloads, Rapid Adoption

2006    SugarCon Conference Established

2007    Sugar Open Source rebranded Sugar Community Edition
        Relicensed to GPLv3

2008    New VCs invest $20M
        Two of the three co-founders leave
        New CEO is Larry Augustin, one of the "founders" of Open Source

2010    Sugar 6 diverges from SugarCE with removal of key features
        Sugar CE is relicensed to AGPLv3

2013    Sugar 7 has no open source counterpart, SaaS only

2014    Sugar announces v6.5 will be the last open source release

2018    Sugar dereleases v6.5 and removes it from availability as far as possible

# The Final Click Of The Ratchet

"To be clear, SugarCRM has ended development and maintenance of the Community Edition. SugarCRM no longer issues new bug fixes, security updates and patches for the Community Edition, and will be soon **removing all Community Edition downloads from SourceForge.net**."

*Clint Oram, 6 Apr 2018*

<u>14 years from inception to discard</u>
*But there were plenty of signs...*

# The Rights Ratchet - Summary

- Start open source for adoption
- Go community for customer onboarding
- Slowly remove rights for non-partners and blame freeloaders
- Go proprietary (or exit) to optimise value

Somewhere on this trajectory today, probably:

Elastic, MongoDB, Redis, Cockroach, Graylog, Confluent, Timescale…

# Counting The Clicks

The Rights Ratchet Model may exhibit these steps (order can vary):

1. Focus is on **adoption** rather than collaboration

2. Licensing that demands a **contributor agreement** so they don't have to submit to it

3. **Differentiated alternative** created as main revenue source

4. Open source code becomes effectively just a **"free trial"** version

5. Assertive **rhetoric** claiming they are as open as they reasonably can be following changes

6. The **license is changed** for reasons not benefiting the community

7. Community no longer recognised and **product now proprietary**

https://meshedinsights.com/2021/02/02/rights-ratchet/

# Spotting The Ratchet

**Everything is likely to be framed as reasonable, fair, positive. Try the scorecard to check the reality:**

- Is the project hosted by a Foundation or a company?

- How was it funded?

- Who owns the trademarks?

- What is the scarcity the company is monetising?

- What is the project license?

- Is there a Contributor Agreement?

- Are the features critical to you open source?

- Is the project realistically forkable?

# Protect Yourself

- Are you sure it's the best choice when this aspect is also considered?

- Make sure the project is forkable
  - Public repository that's used for actual development
  - Keep a sync copy yourself, constantly
  - Check it can be built without trademarks
  - Actually, check it can be built!
  - Keep track of other large users

- Has there been a viable fork already?

- Consider if the CLA can be avoided
  - Is there a license in = license out option?
  - Can your legal team negotiate a better arrangement?

- Advocate for "Permission In Advance" rights while you have leverage

https://meshedinsights.com/2021/03/02/permission-beyond-licensing/

# Summary

- The Rights Ratchet is (almost) as old as open source

- Projects clicking may prove unavoidable

- Invest in those that don't ratchet

- Take precautions or use alternatives

# Thanks for your attention!

This presentation was written and delivered by Simon Phipps

simon@meshedinsights.com

https://meshedinsights.com

Want to discuss this? Book via https://clarity.fm/webmink