

FOSS Backstage – 13.6.2018

The Hateful 20

Lars Francke – Partner and Co-Founder @ OpenCore



About Me – Lars Francke

- Partner & Co-Founder at OpenCore
- Before that: Freelance Hadoop Consultant
- Hadoop since 2008/2009
- Apache Committer & Member: HBase, Hive, ORC

- Contact
 - lars.francke@opencore.com
 - [@lars_francke](https://twitter.com/lars_francke)

Disclaimer

Disclaimer

- Highly subjective opinions
 - Subjectively: I have a feeling things are slowly getting better with some projects
- Focused on the Apache ecosystem
- I'm guilty of some of the things I mention as well
- Don't take everything I say entirely serious

Despite my ranting: I still love all of this, otherwise I wouldn't work in this field and care so much

I'm sorry to be standing between you and lunch.

But: It's only 20 more minutes!

What's this all about?





NEEDO

CLOSED



Main Entry: fauxpen source

Pronunciation: \fō-pən sōrs\

Function: noun

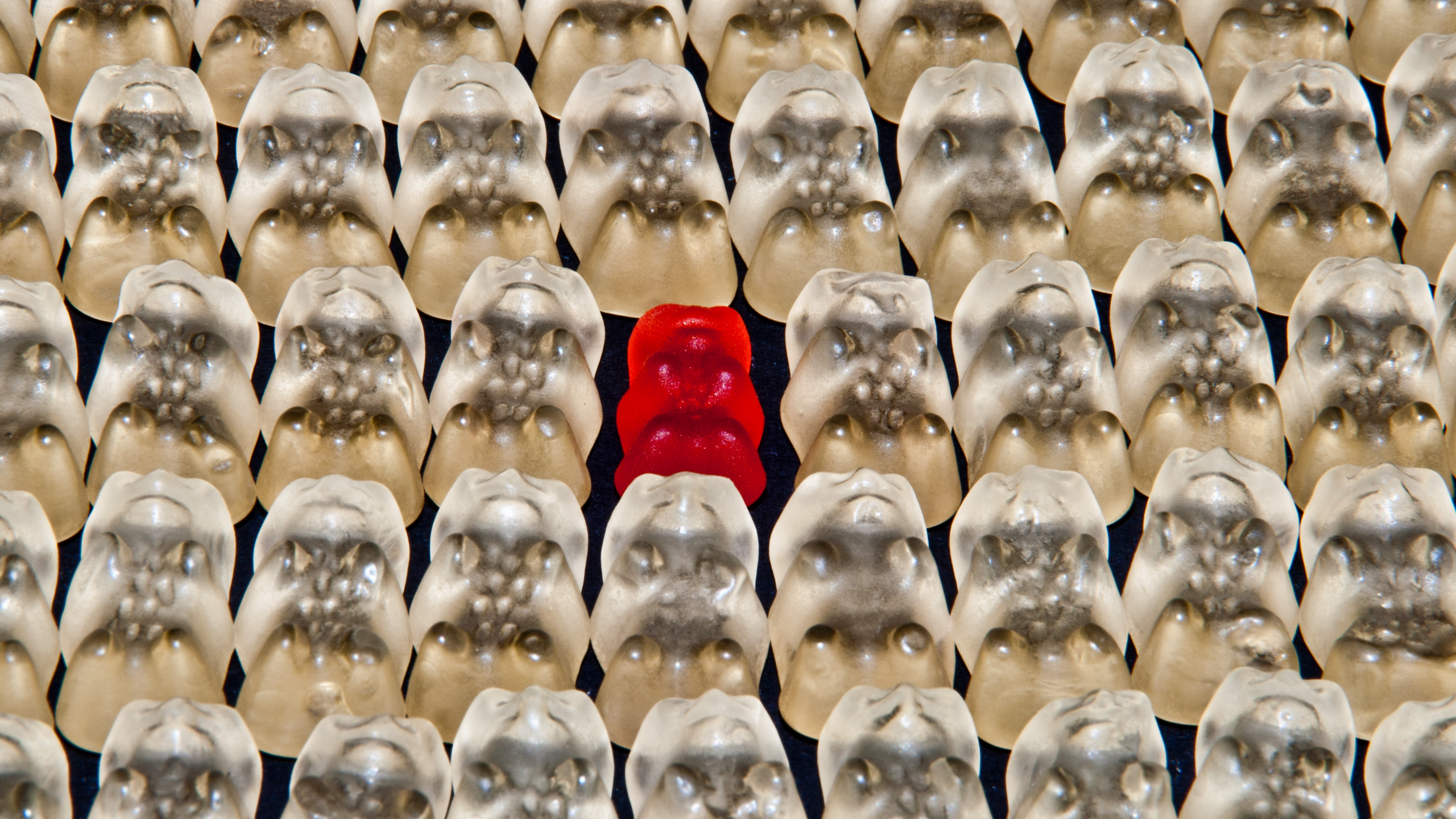
Etymology: a term invented by [Phil Marsosudiro](#) at a dinner party in North Carolina

Date: 2 May 2009

: A description of software that claims to be open source, but lacks the full freedoms required by the [Open Source Definition](#).

synonyms: see [open core](#), [neo-proprietary](#)

antonyms: see Linux [kernel](#), [Apache](#), [OpenNMS](#)







Closing thoughts

- Recently lots of discussions on project mailing lists about the value of non-code contributions and what else should gain someone committership
- Training in the "The Apache Way" (e.g. Hortonworks) is a good thing
- Some contributors are hard to deal with
- Every project & contributor is different
 - →Hard to find rules that work for everyone

The End!

Thank you for listening!

Questions?

lars.francke@opencore.com

The following slides

- The following slides weren't part of the presentation
- They give a textual summary of most/all the points I only talked about

Documentation & None code contriibs

- Documentation often lacking
- But contributions not valued or rewarded as often
- Committership should be for things other than code as well
 - As a form of recognition and encouragement
- Engagement on mailing lists, Slack, IRC, Stack Overflow, Presentations at conferences, books, articles etc.

- But: Is being recognized and addressed by some projects, hard to identify valuable people though as not focused purely on the project infrastructure

Clean code

- Code quality often not valued
- Patches focusing on code quality and style alone often not accepted
 - They should be, bad code is just as big a bug as broken code
- Undocumented, inconsistent code raises the bar for newcomers to get started

- But: Code cleanliness hard to maintain, strict enforcement might discourage some contributions

Closed Open Source

- When an Open Source project is used for "marketing" or "whitewashing" but processes are really more like a closed source tool
- Discussions behind closed doors
 - Often informal and not intentional
- JIRAs being opened, commented on, reviewed, committed and closed within five minutes

- But: Makes a project agile while still keeping some of the open source benefits, fast reaction times

Outsiders

- Lots of projects "controlled" by vendors
- Hard for outsiders to join & get recognized
 - Especially non-US (or even non-valley) people due to lack of personal access
- Code often not in a state that makes it self-explanatory
- Not exposed to company-internal deadlines or goals
- Might not share the same (implicit) view on project management, goals, code style etc. as the core team

No Project Management

- No one to talk to when unsure about direction of the project
- Business investing multi-million dollar sums without anyone to guarantee direction
- Today this, tomorrow that
 - e.g. dropping support for certain dependencies (Java, Hadoop, Spark) might be due to lack of resources or willingness rather than business decisions
- Project processes not clearly documented or followed

- But: No project management can be a good thing, not tied to strict goals, rules etc.

JIRA & Patches

- Smaller contributions from newcomers often just "die" in JIRA due to lack of reviews
- Mentors needed for people getting started
- Newcomers often held to higher standards than established people
- "Bikeshedding": Huge patches with blatant flaws (e.g. no/disabled tests, hundreds of checkstyle warnings etc.) being accepted while small ones get denied because of a single wrong indentation
- Contributors lose interest if it becomes too hard to contribute