

Community Guide to Contributing to Mature Software Products

Michael Kromer
VP Technology & Architecture

\$ whoami

- Mike, Geek.
- First computer with 6 (C64)
- Background in US military background (Scalability, FUBAR-safe)
- VP Technology & Architecture (Kopano)
- <http://medozas.de/vita.html>
- openSUSE Member und Committer
- Committed code to Kopano, the Linux kernel, Asterisk, LibreOffice, KVM, Virtualbox, Nomachine NX, X2Go, OpenSER, Chromium, USBoverIP,

There is no **one** community.
Every software community works a little or more different.

Examples

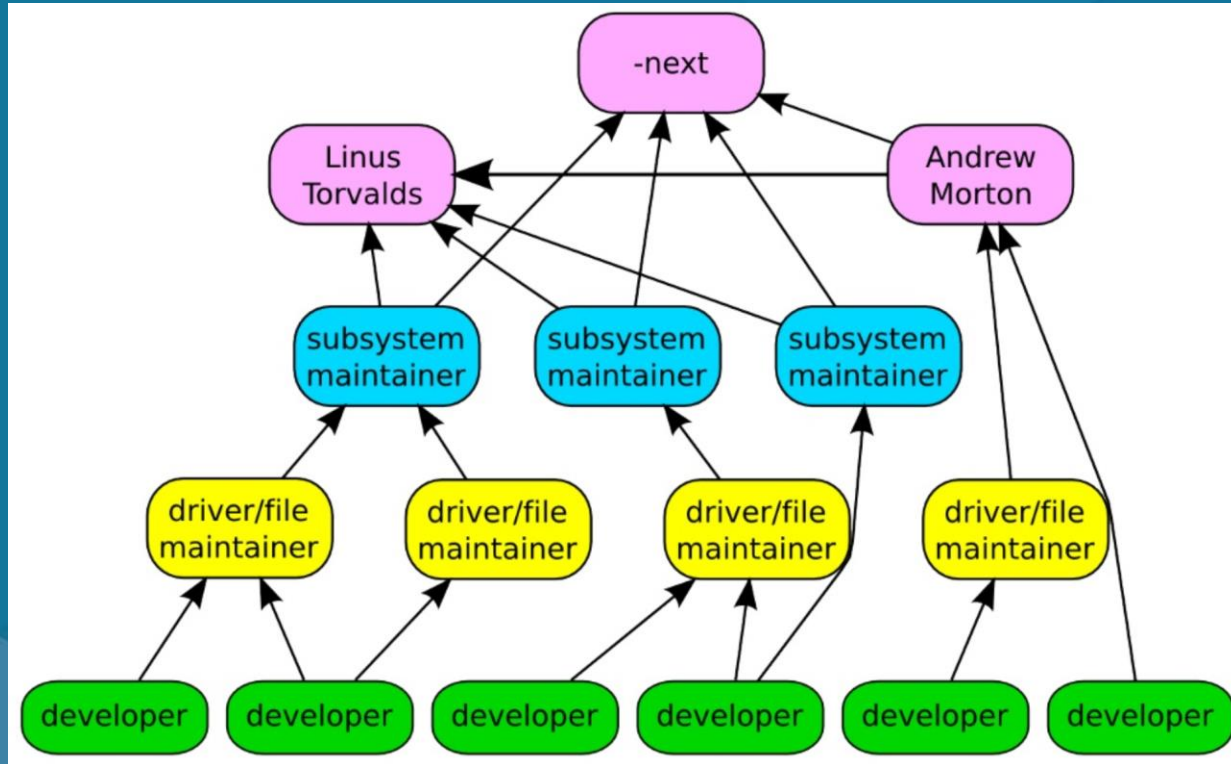
- Linux Kernel
- Iridium Browser
- Kopano
- openSUSE / SLE

Different organisational structures

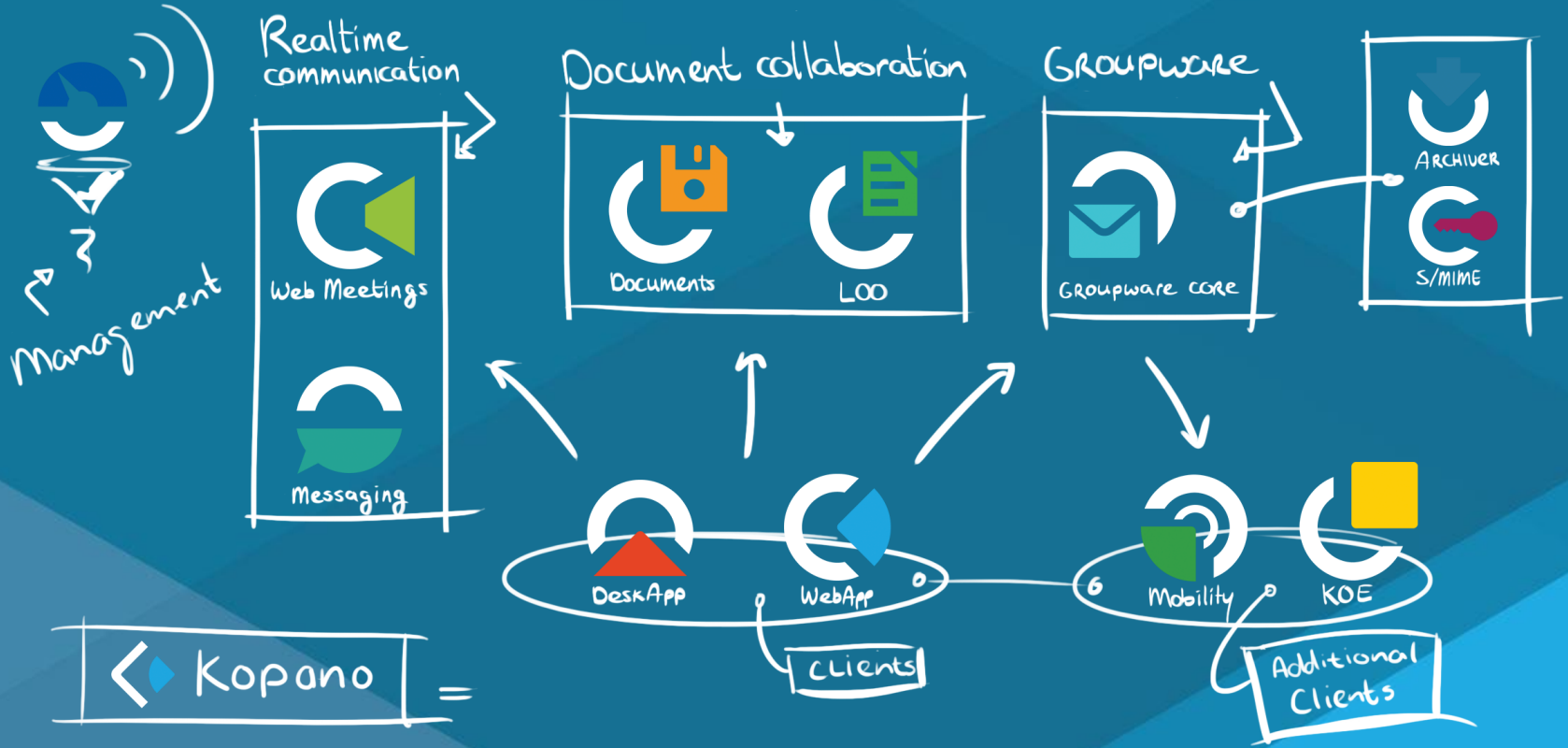
- Linux Kernel → „demand-oriented“ with chain of trust
- Iridium Browser → „project-oriented“ with a specific goal
- openSUSE → open community with commercial support
- SLE → company-driven, customer & market oriented
- Kopano → „hybrid“ community with commercial backing

Example of code & project structure

Linux



Kopano



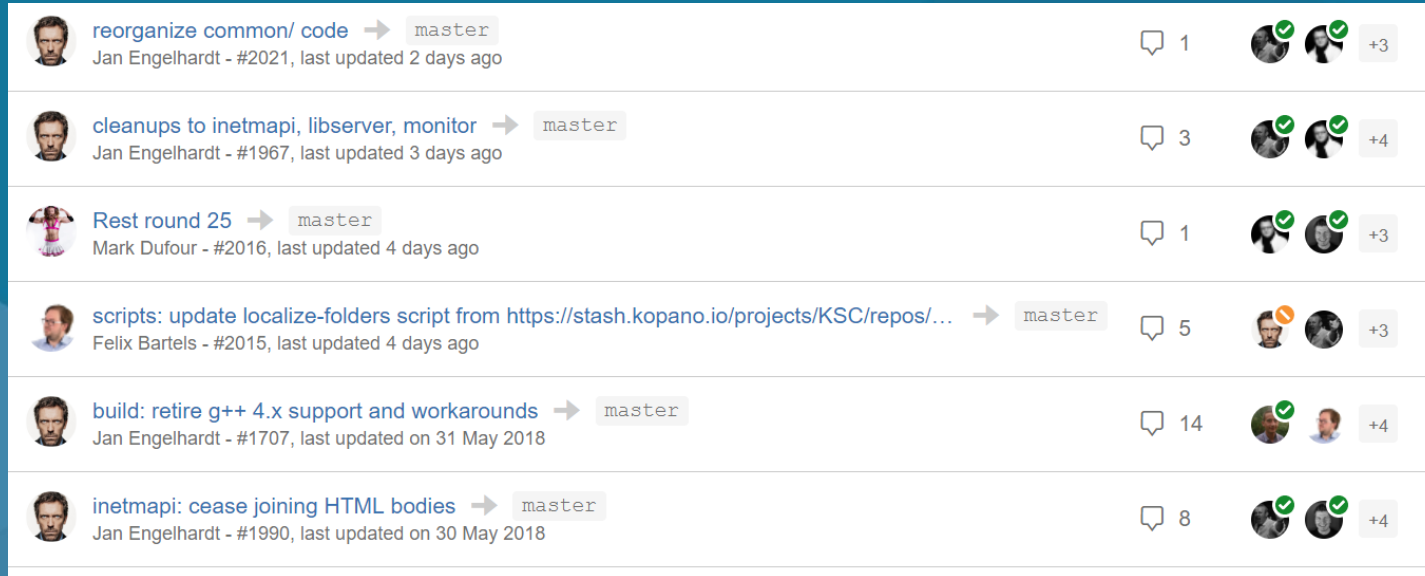
Lifecycle overview of a mature software product



From code to quality code

Review

- Reviews by developers in that area (merit & experience based)



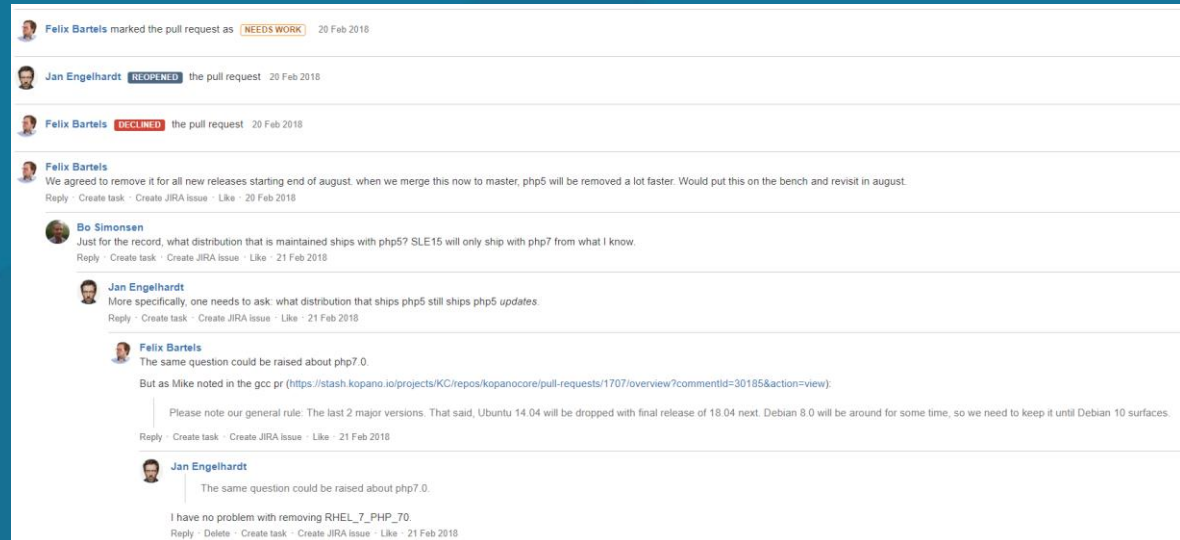
A screenshot of a GitHub pull request list. The list contains six entries, each with a user profile picture, a commit title, a branch name, a comment count, and a number of approvals. The entries are:

Author	Commit Title	Branch	Comments	Approvals
Jan Engelhardt	reorganize common/ code	master	1	+3
Jan Engelhardt	cleanups to inetmapi, libserver, monitor	master	3	+4
Mark Dufour	Rest round 25	master	1	+3
Felix Bartels	scripts: update localize-folders script from https://stash.kopano.io/projects/KSC/repos/...	master	5	+3
Jan Engelhardt	build: retire g++ 4.x support and workarounds	master	14	+4
Jan Engelhardt	inetmapi: cease joining HTML bodies	master	8	+4

From code to quality code

Review

- Code quality
- Potential side effects
- Style guidelines

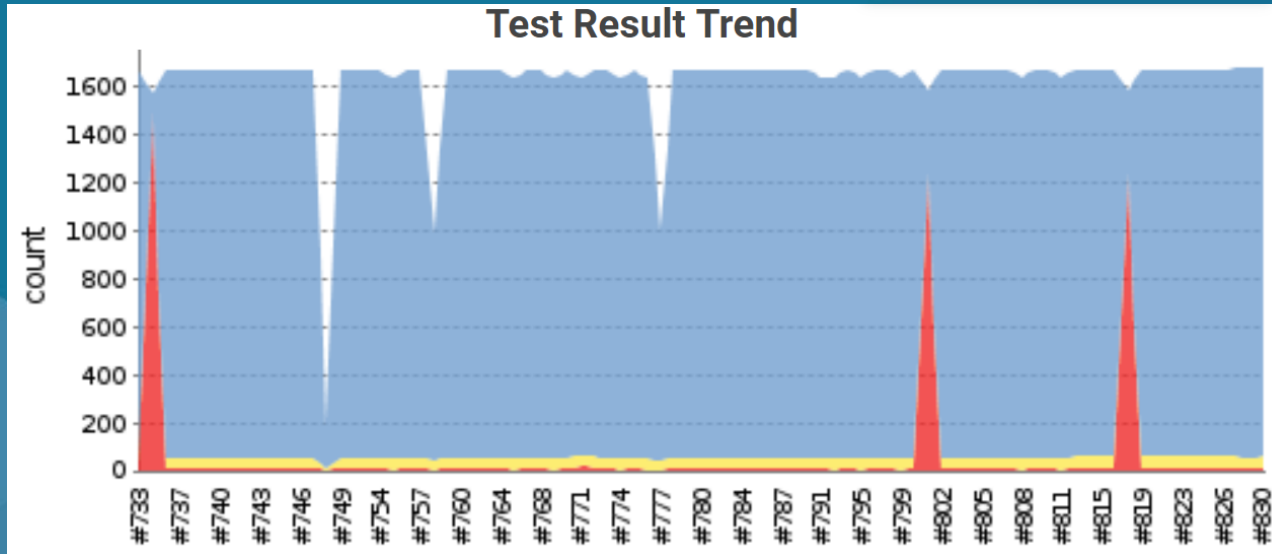


The screenshot shows a GitHub pull request discussion thread with the following content:

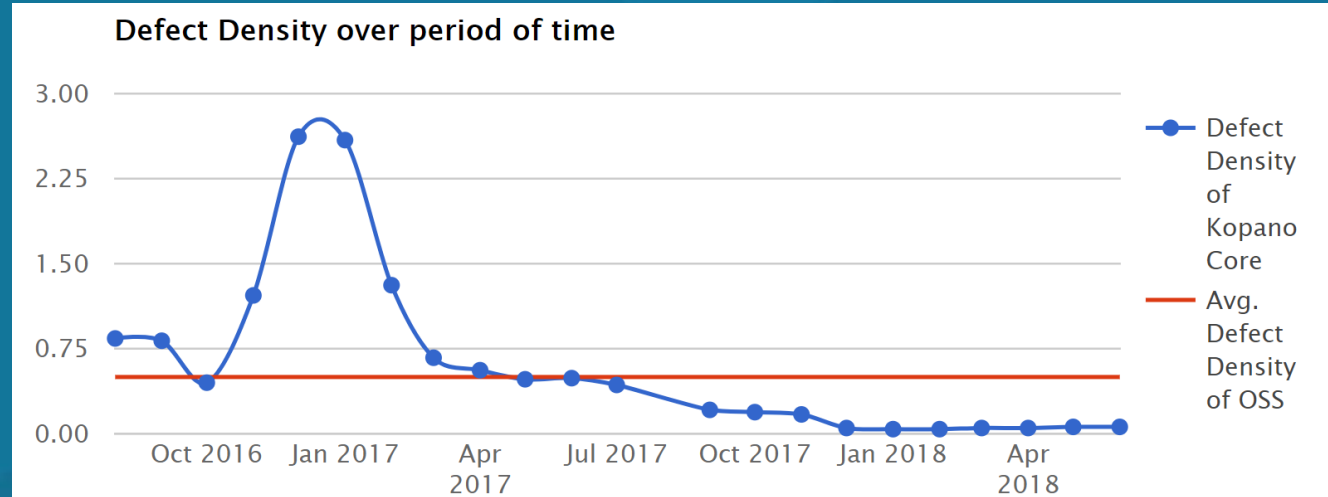
- Felix Bartels** marked the pull request as **NEEDS WORK** · 20 Feb 2018
- Jan Engelhardt** **REOPENED** the pull request · 20 Feb 2018
- Felix Bartels** **DECLINED** the pull request · 20 Feb 2018
- Felix Bartels**
We agreed to remove it for all new releases starting end of august, when we merge this now to master, php5 will be removed a lot faster. Would put this on the bench and revisit in august.
Reply · Create task · Create JIRA issue · Like · 20 Feb 2018
- Bo Simonsen**
Just for the record, what distribution that is maintained ships with php5? SLE15 will only ship with php7 from what I know.
Reply · Create task · Create JIRA issue · Like · 21 Feb 2018
- Jan Engelhardt**
More specifically, one needs to ask: what distribution that ships php5 still ships php5 updates.
Reply · Create task · Create JIRA issue · Like · 21 Feb 2018
- Felix Bartels**
The same question could be raised about php7.0.
But as Mike noted in the gcc pr (<https://stash.kopano.io/projects/KC/repos/kopancore/pull-requests/1707/overview?commentId=30185&action=view>).
Please note our general rule: The last 2 major versions. That said, Ubuntu 14.04 will be dropped with final release of 18.04 next, Debian 8.0 will be around for some time, so we need to keep it until Debian 10 surfaces.
Reply · Create task · Create JIRA issue · Like · 21 Feb 2018
- Jan Engelhardt**
The same question could be raised about php7.0.
I have no problem with removing RHEL_7_PHP_70.
Reply · Delete · Create task · Create JIRA issue · Like · 21 Feb 2018

From code to quality code CI

- Automated quality checks (unit tests, merge tests, sanity checks, linting checks, memory leak checks)



From code to quality code CI



- Regressions?
- Instability?
- Degration of (scalability|performance|interoperability)?
- Upgrade-safe?

From code to quality code

QA

- Verification of all checks (Is Test-covered?)
- Manual check if „issue“ is solved
- Documentation changes



From code to quality code

Release

- Backwards incompatibilities? (API/ABI)
- Behavioral changes?
- Default settings?

<input type="checkbox"/>	All commits merged to the release branch
<input type="checkbox"/>	Commits announced at the forums https://forum.kopano.io/topic/901/z-push-2-4-x-beta-pre-final-updates/
<input type="checkbox"/>	Create a tag on the master (final release) or release branch (beta release)
<input type="checkbox"/>	Check if the builds are available at http://repo.z-hub.io/z-push:final/ or http://repo.z-hub.io/z-push:beta/ If not, check Release procedures#Buildingpackages page how to build packages
<input type="checkbox"/>	Update the ZP-1335 - Maintain 2.4 release branch IN PROGRESS
<input type="checkbox"/>	Create release notes for the forums
<input type="checkbox"/>	Announce the release at https://forum.kopano.io/category/26/general-usage-of-z-push/
<input type="checkbox"/>	For final release do blog post at http://z-push.org/category/news/
<input type="checkbox"/>	For final release update information at http://z-push.org/download/
<input type="checkbox"/>	Update releases page at https://jira.z-hub.io/projects/ZP?selectedItem=com.atlassian
<input type="checkbox"/>	Update Version Lifecycle page
<input type="checkbox"/>	Update Roadmap page
<input type="checkbox"/>	For final release email Toos and Bob, cc Felix and Z-Push devs with the link to the release
<input type="checkbox"/>	Announce release at https://mattermost.kopano.com/core/channels/kopano-releases
<input type="checkbox"/>	For final release Send an email to the Z-Push mailing list (Z-Push@lists.inai.de)

Kopano

Questions?