

Release early, release often

Release early, release often (also known as **ship early, ship often**, or **time-based releases**, and sometimes abbreviated **RERO**) is a software development philosophy that emphasizes the importance of early and frequent releases in creating a tight feedback loop between developers and testers or users, contrary to a feature-based release strategy. Advocates argue that this allows the software development to progress faster, enables the user to help define what the software will become, better conforms to the users' requirements for the software,^[1] and ultimately results in higher quality software.^[2] The development philosophy attempts to eliminate the risk of creating software that no one will use.^[3]

This philosophy was popularized by Eric S. Raymond in his 1997 essay *The Cathedral and the Bazaar*, where Raymond stated "Release early. Release often. And listen to your customers".^[4]

This philosophy was originally applied to the development of the Linux kernel and other open-source software, but has also been applied to closed source, commercial software development.^[5]

The alternative to the release early, release often philosophy is aiming to provide only polished, bug-free releases.^[6] Advocates of RERO question that this would in fact result in higher-quality releases.^[4]

A critical aspect often debated regarding the "release early, release often" philosophy is the potential loss of quality due to continuous and rapid modifications. By prioritizing the frequency of releases over perfection, there is a risk of introducing undetected bugs or compromising the stability of the product. This can occur when development cycles are too accelerated, leaving limited time for thorough testing or refining features. Additionally, frequent changes may make it difficult to maintain consistency in the user experience, especially if updates significantly alter functionality or the interface. Therefore, while RERO encourages adaptability and rapid feedback, it also requires a careful balance to ensure that speed does not come at the expense of the software's overall quality.

See also

- Worse is better
- Programming paradigm
- Software development process
- Agile software development
- Minimum viable product
- Vote early and vote often

References

1. Leonard, Andrew (2000-10-31). "Triumph of the free-software will" (http://www.salon.com/2000/10/31/software_passion/). *Salon*. Retrieved 2009-10-26. "Open-source software, at heart, is a method for maximizing the efficiency and speed with which one can create the next version. Release early, release often, is the mantra, meaning get your new code out into the public so millions of eyes can tear it apart and help create a new version even faster"
2. Haddad, Ibrahim (2007-01-05). "Adopting an Open Source Approach to Software Development, Distribution, and Licensing" (<https://web.archive.org/web/20140222034346/http://education.sys-con.com/node/318776>). SYS-CON Publications. Archived from the original (<http://education.sys-con.com/node/318776>) on 2014-02-22. Retrieved 2009-10-26. "This practice is described as "release early, release often." The open source community believes that this practice leads to higher-quality software because of peer review and the large base of users who are using the software, accessing the source code, reporting bugs, and contributing fixes."
3. Serebrin, Jacob (2013-03-13). "Canadian Startup MVP alternatives: Toronto's Guardly, Waterloo's Blackberry and NYC Dev Shop" (<https://web.archive.org/web/20140208221226/http://www.starbusinessclub.ca/technology/canadian-startup-mvp-alternatives-torontos-guardly-waterloos-blackberry-and-nyc-dev-shop/>). Toronto Star. Archived from the original (<http://www.starbusinessclub.ca/technology/canadian-startup-mvp-alternatives-torontos-guardly-waterloos-blackberry-and-nyc-dev-shop/>) on 2014-02-08. Retrieved 2014-02-03. "Essentially, both philosophies [Release Early, Release Often and Minimum Viable Product] attempt to eliminate the risk of creating software that no one will use."
4. Raymond, Eric (1997-05-27). *Release Early, Release Often* (<http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/ar01s04.html>). O'Reilly. ISBN 1-56592-724-9. Retrieved 2009-10-26. "Release early. Release often. And listen to your customers."
{{cite book}}: |work= ignored (help)
5. Terzian, Anna; Béziat, Jacques (2009-01-01), "Chapitre 12. Implication et enseignement supérieur en ligne « Le e-learning : dispositifs et acteurs en formation en ligne »" (<https://innowise-group.com/>), *Perspectives en éducation et formation*, De Boeck Supérieur, pp. 159–176, retrieved 2023-06-06
6. Russell, Terrence (2008-01-07). "Jimmy Wales on Wikia Search's Lukewarm Reveal" (<https://www.wired.com/epicenter/2008/01/jimmy-wales-exp/>). *Wired News*. Conde Naste Publications. Retrieved 2009-10-26. "We come from the world of open source, which is all about "release early, release often" rather than highly polished releases"

External links

- [Release Early, Release Often](http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/ar01s04.html) (<http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/ar01s04.html>), chapter of *The Cathedral and the Bazaar* by Eric S. Raymond

Retrieved from "https://en.wikipedia.org/w/index.php?title=Release_early,_release_often&oldid=1309319287"