

# ソフトウェアライセンスのメモ

2024/01/06

- 著作権（コピーライト（Copyright）と同義）
- 利用許諾（ライセンス（License）と同義）  
ソフトウェアの著作権（コピーライト）を持つものは、そのソフトウェアのユーザへの利用許諾（ライセンス）を自由に決めることができる。
- デュアル・ライセンス  
複数の利用許諾（ライセンス）でプログラムをユーザに提供することを、デュアル・ライセンス と呼ぶ。例えば、MySQL は デュアル・ライセンス していて、コミュニティ版と商用版がある。X11 も有名。GitLab CE/EE も。
- 派生物（二次的著作物（Derivative works）と同義）  
プログラムを改変し再配布する場合、元のプログラムを原著物（Original work）、改変後の著作物を派生物と呼ぶ。
- パブリックドメイン  
ソフトウェアをパブリックドメイン（公共）にする、ということは、著作権者が著作権を放棄し、誰でもどのような利用も許諾される状態にする、ということ。
- フリーソフトウェア（Free Software）
- オープンソースソフトウェア（Open-source Software（OSS））
  - <https://www.gnu.org/philosophy/free-sw.html>  
タダ（無償）という意味ではない、真に自由な（フリーな）ソフトウェアとは何か、という思想。ユーザがプログラム（開発）の恩恵を得られる権利を維持することを自由と呼ぶ、というような思想。ソースが無いと過去の開発の恩恵を得られないので、ソースはオープンにしよう、という意味で、フリーソフトウェアとオープンソースソフトウェアの考え方はほぼ同義、ほぼ同一。
- コピーレフト
  - <https://www.gnu.org/licenses/copyleft.ja.html>  
ユーザがプログラムを自由に利用できる権利を守るために、派生物の利用許諾にも強い制限を課し、1) ソースを公開しなければならない、2) 同じ利用許諾を使わなければならない、とする思想。フリーソフトウェア/オープンソースソフトウェア の考えの根底にある思想。copyright – all rights reserved.（著作権 – 全ての権利は留保されている）を揶揄した、copyleft – all rights reversed.（コピーレフト – 全ての右は逆転されている）に由来する。ユーザの利用できる権利を守るため、該当プログラムを利用した派生物は全て、ソースを公開するなどの同じ制限と利用許諾にしなければならない。GPL（GNU Public License）が有名。ユーザの権利を守ろうとする余り、プログラムの派生物にも同じコピーレフト性という強い制限を求めるため、GPL 汚染やライセンス感染と呼ばれる。開発物のソースをすべて

公開しなければならないという制限が、多くのビジネスモデルと適合しない、と言われる。

- Permissive ライセンス

- <https://opensource.org/faq/#permissive>

GPL のような制限の強いコピーレフトライセンスに対して、MIT や BSD 2/3-Clause の寛容なライセンスは Permissive ライセンスと呼ばれる。このライセンスのソフトウェアも、正しくフリーソフトウェア/オープンソースソフトウェアであるとされる。著作権表示を消すな、大学名を販促に使うな、というような緩い制限の 2/3 項目を守れば、商用や改変・再配布を含め、どのような利用もできるライセンス。Apache ライセンスもこれ。

- Selling Exceptions

- <https://www.gnu.org/philosophy/selling-exceptions.en.html>

真に自由なコピーレフトライセンスを使って、ビジネス（商用利用）ができるようにしよう、という、GNU の祖ストールマン (RMS) の例外規定。GNU より前からある X11 を許すためには、という記述が見られる。

- Commons Clause

- <https://commonsclause.com/>

オープンソースライセンスの前段にこれを付ければ、商用利用だけを禁止できるのでは？という提案。かなり批判された模様。

- Open-core モデル

- [https://en.wikipedia.org/wiki/Open-core\\_model](https://en.wikipedia.org/wiki/Open-core_model)

デュアル・ライセンスでビジネス・商用利用もしつつ、ベース部分をオープンソースソフトウェアとするビジネスモデル。Kafka, Cassandra, Oracle/MySQL, Eucalyptus, GitLab, Redis などが該当するらしい。

- Source-available Software

- [https://en.wikipedia.org/wiki/Source-available\\_software](https://en.wikipedia.org/wiki/Source-available_software)

ソースを公開しているが、フリーソフトウェア/オープンソースソフトウェアにそぐわないライセンスでソフトウェア開発・提供しているモデル。Elastic が Open-core からこちらに移行した模様。Redis の plug-in 部分もこのモデルのライセンス (Redis Source Available License)。HasiCorp/Terraform もこのモデル (Business Source License (BSL))。MongoDB の SSPL もソース公開だがフリー・オープンソースでない商用ライセンスであり、このモデル。

- 伽藍とバザール

- <https://www.aozora.gr.jp/cards/000029/card227.html>

荘厳な伽藍を建築するのではなくて、バザールのように人々に勝手にやらせよう、というソフトウェア開発の方針の思想。FRRouting (Free-range (放し飼いの) Routing) はバザールと同じような思想で開発されている。

- LGPL

- <https://www.gnu.org/licenses/lgpl-3.0.html>

Link するだけのプログラムには GPL を求めないような、Lesser (劣等) GPL。ライブラリのために生まれた。

- Creative Commons

- <https://creativecommons.jp/licenses/>

文書のためのライセンス体系、部品になっていて組み合わせで使える。CC BY-NC-ND（BY: 著作権表示、NC: 非商用、ND: 改変禁止）など。FAQ で、ソフトウェアに CC を付与することは可能だが、お勧めしない、と書いてある。

- PolyForm

- <https://polyformproject.org/>

ソフトウェアに適応できる CC のようなものらしい。非商用、とかある模様。