CENDI 2009-1 Rev 2

Frequently Asked Questions about Copyright and Computer Software

Issues Affecting the U.S. Government with Special Emphasis on Open Source Software

Prepared by
CENDI Copyright Working Group

Compiled and
edited by
Vicki Allums
Defense Acquisition University
and
Nancy Kremers
Department of the Air Force

November 1, 2009
Revised August 2020

FREQUENTLY ASKED QUESTIONS ABOUT COPYRIGHT AND COMPUTER SOFTWARE: ISSUES AFFECTING THE U.S. GOVERNMENT WITH SPECIAL EMPHASIS ON OPEN SOURCE SOFTWARE

Prepared by

CENDI Copyright Working Group

DISCLAIMER

THIS DOCUMENT DOES NOT CONSTITUTE LEGAL ADVICE AND SHOULD NOT BE CONSTRUED OR USED AS SUCH. For specific questions related to the use of proprietary and open source computer software, licensing, and copyright issues, consult the appropriate program office and your agency's Office of the General Counsel. Consult CENDI's "Frequently Asked Questions about Copyright" for general information on copyrighted and U.S. Government works.

Purpose and Use of This Document

This document provides general guidance on a special category of copyrighted works— computer software—and includes a detailed discussion of open source software. Federal agencies are increasingly supporting the use and acquisition of open source software as an alternative to proprietary software in their information technology programs. It is hoped that this Frequently Asked Questions (FAQ) document will serve as a useful resource for contracting officers, program managers, librarians, information center staff, and attorneys.
The document was revised in 2019 to reflect changes in federal source code policy. In August 2016, the Office of Management and Budget (OMB) published Memorandum M-16-21, which requires that federal agencies share new custom-developed code they create or procure for broad reuse across the federal government; and to release at least 20% of new custom-developed code to the public as OSS. OMB also established principles and guidelines for federal agencies to engage with the open source community to collaborate on code development and improvement.

Copyright Notice

Notice of Change

The information presented in this FAQ is subject to changes enacted by U.S. Government policies, legislation and case law.

CENDI is an interagency cooperative organization composed of the scientific and technical information (STI) managers from the Departments of Agriculture, Commerce, Defense, Energy, Education, the Environmental Protection Agency, Health and Human Services, Interior, the National Aeronautics and Space Administration, the Government Publishing Office and the National Science Foundation. CENDI's mission is to help improve the productivity of federal science- and technology-based programs through the development and management of effective scientific and technical information support systems. In fulfilling its mission, CENDI member agencies play an important role in helping to strengthen U.S. competitiveness and address science- and technology-based national priorities.

Table of Contents

CENDI Copyright Working Group

Contributing Members:

Bill Adams (Army); Scott Albright (EPA); Vicki Allums (DAU); Jane Barrow (NAVSEA); Dale Berkley (NIH); Gary Borda (NASA); Cindy Clark (NIH); Christopher Cole (NAL); Geoffrey Cooper (EPA); Charles Ducker (DoT); Linda Field (DOE); Kathryn Funk (NIH/NLM); Rebecca Goodwin (NIH/NLM); Courtney Graham (NASA); Richard Gray (DoD); Phil Greene (USMC); Gail Hodge (CENDI); Laura Jennings (NGA); Rob Kasunic (LoC); Flayo Kirk (MDA); Bonnie Klein (DTIC); Nancy Kremers (USAF); Richard Lambert (NIH); Jeffrey Landou (NARA); Jan McNutt (NASA); Jeffrey Moore (AFRL); Hope O'Keeffe (LoC); Dina Paltoo (NLM); Vinit Patel (DoE); Don Pollack (DLA/DTIC); Timothy Slabouz (USMC); John Raubitschek (Army), Vakare Valaitis (DTIC); Damien Walsh (OCJCS); George Winborne (Army)

1 Glossary of Terms

1.1 Abbreviations and Acronyms

BSD	Berkeley Software Distribution
CC	Creative Commons
COTS	Commercial-off-the-shelf
CRADA	Cooperative Research and Development Agreement
DFARS	Defense Federal Acquisition Regulation Supplement
DISA	Defense Information Systems Agency
DOC	Department of Commerce
DoD	Department of Defense
DOE	Department of Energy
DoN	Department of the Navy
EPA	Environmental Protection Agency
FAR	Federal Acquisition Regulation
FISMA	Federal Information Security Management Act
FOIA	Freedom of Information Act
FSF	Free Software Foundation
GPO	Government Printing Office
GNU GPL	General Public License
LGPL	Lesser General Public License
HHS	Department of Health and Human Services
MPL	Mozilla Public License
NARA	National Archives and Records Administration
NASA	National Aeronautics and Space Administration
NIH	National Institutes of Health
NOSA	NASA Open Source Agreement
NSF	National Science Foundation
OMB	Office of Management and Budget
OSA	Open Systems Architecture
OSI	Open Source Initiative
OSS	Open Source Software
USC	United States Code
USPTO	US Patent and Trademark Office

1.2 Definitions

While the following terms may have more than one generally accepted meaning, as used in these FAQs (Frequently Asked Questions), they are defined as follows.

Computer Program means a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result. (See 17 USC § 101.)

DFARS 252.227.7014 (3) defines ***Computer program*** as "a set of instructions, rules, or routines, recorded in a form that is capable of causing a computer to perform a specific operation or series of operations."

Computer Software or ***software*** means one or more computer programs.

FAR 2.101 defines ***Computer software*** as "(i) Computer programs that comprise a series of instructions, rules, routines, or statements, regardless of the media in which recorded, that allow or cause a computer to perform a specific operation or series of operations; and (ii) Recorded information comprising source code listings, design details, algorithms, processes, flow charts, formulas, and related material that would enable the computer program to be produced, created, or compiled. [ ] Does not include computer databases or computer software documentation."

DFARS 252.227.7014 defines ***Computer software*** as "computer programs, source code, source code listings, object code listings, design details, algorithms, processes, flow charts, formulae, and related material that would enable the software to be reproduced, recreated, or recompiled. Computer software does not include computer databases or computer software documentation."

Commercial Computer Software, as defined in the DFARS and FAR, means software developed or regularly used for non-governmental purposes, which has been sold, licensed or leased to the public or is a commercial item. (See DFARS 252. 227.7014 (a) (i) and FAR 2.101.) Open Source Software is commercial computer software licensed under a licensing scheme that provides broad rights to modify and redistribute the original source code and, sometimes, any distributed modified versions (e.g., derivative works). (See FAQ Section 3.1.)

Derivative Work means a work that is based upon one or more preexisting works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which a work may be recast, transformed, or adapted. A work consisting of editorial revisions, annotations, elaborations, or other modifications, which, as a whole, represent an original work of authorship, is a "derivative work." In the computer industry, a second version of a software program is generally considered a derivative work based upon the earlier version. (See 17 USC § 101.) The term "derived work" is often used in commercial parlance to mean "derivative work."

Defense Federal Acquisition Regulation Supplement (DFARS) means the supplement to the Federal Acquisition Regulations used by the Department of Defense to purchase goods and services.

Executable Code means a subroutine, method, procedure, or subprogram of a larger program that performs a specific task and can operate relatively independent of the remaining code. It can be self-contained or call upon other code to execute (take a specific action).

Federal Acquisition Regulation (FAR) means the regulation established to codify uniform policies for acquisition of supplies and services by federal executive agencies. It is issued and maintained jointly, pursuant to the Office of Federal Procurement Policy (OFPP)

Reauthorization Act, under statutory authorities granted to the Secretary of Defense (DoD), Administrator of General Services (GSA), and the Administrator, National Aeronautics and Space Administration (NASA). The official FAR appears in the Code of Federal Regulations at 48 CFR Chapter 1. The FAR applies to procurement contracting only; i.e., contracts to procure goods and services primarily for the benefit of the federal government. Other, very different laws and regulations apply to non-procurement award instruments, such as grants, cooperative agreements, "other transactions" agreements, CRADAs, and international agreements. Note also that a number of government agencies use an agency-specific version, or supplement, to the FAR; the DFARS, defined above, is one example. Even within government procurement contracting, there may be important substantive differences between the FAR and an agency supplement, so it is important to identify which acquisition regime is applicable to any particular transaction.

Object Code means computer program code that is written in machine-readable language.

Open Architecture is a technical architecture that adopts open standards supporting a modular, loosely coupled and highly cohesive system structure that includes publishing of key interfaces within the system and full design disclosure, incorporating appropriate considerations for reconfigurability, portability, maintainability, technology insertion, vendor independence, reusability, scalability, interoperability, upgradeability, and long-term supportability.[1]

Open Systems Architecture (OSA) is a system that employs modular design, uses widely supported and consensus based standards for its key interfaces, and has been subjected to successful validation and verification tests to ensure the openness of its key interfaces. An open architecture is defined as a technical architecture that adopts open standards supporting a modular, loosely coupled and highly cohesive system structure that includes publishing of key interfaces within the system and full design disclosure. The key enabler for open architecture is the adoption of an open business model, which requires doing business in a transparent way that leverages the collaborative innovation of numerous participants across the enterprise permitting shared risk, maximized asset reuse and reduced total ownership costs. The combination of open architecture and an open business model permit the acquisition of Open Systems Architectures that yield modular, interoperable systems allowing components to be added, modified, replaced, removed and/or supported by different vendors throughout the life cycle in order to drive opportunities for enhanced competition and innovation.
The following are the core principles of the Open Systems Architecture approach:

1. Modular designs with loose coupling and high cohesion that allow for independent acquisition of system components, i.e., composability;
2. Continuous design disclosure and appropriate use of data rights allowing greater visibility into an unfolding design and flexibility in acquisition alternatives;
3. Enterprise investment strategies that maximize reuse of system designs and reduce total ownership costs (TOC);
4. Enhanced transparency of system design through government, academia, and industry peer reviews;

[1] Open Systems Architecture Contract Guidebook for Program Managers, Version 1.1, May 2013
https://www.dau.edu/cop/pm/DAU%20Sponsored%20Documents/Open%20System%20Architecture%20(OSA)%20Contract%20Guidebook%20for%20Program%20Managers-version%201.1-%20June%20%202013.pdf?Web=1

5. Competition and collaboration through development of alternative solutions and sources; and

6. Analysis to determine which components will provide the best return on investment (ROI) to OSA, i.e., which components will change most often due to technology upgrades or parts obsolescence and have the highest associated cost over the life cycle.

Achievement of these six principles requires an affirmative answer to a fundamental question: Can a qualified third party add, modify, replace, remove, or provide support for a component of a system, based on open standards and published interfaces for the component of that system?[2]

Open Source License is a license to software that provides the licensee the freedom to use the software for any purpose, to modify the software, and to redistribute copies of the original or modified software without payment of royalties. In order to provide the user these freedoms, open source licenses require that the user have access to and use of the software source code.

Open Source Software software for which the human-readable source code is available for use, modification, and re-distribution by the users of that software.

Permissive Open Source Licenses allow distribution of the original and derivative works of the open source software under different terms than the original open source license. Thus, derivative works can be licensed as proprietary software, and the original open source software can be incorporated into proprietary software.

Proprietary software means software in which the owner reserves certain rights that limit the way in which the software may be used by others. Many proprietary software products are commercial computer software.

OMB M 16-21, "Federal Source Code Policy, Achieving Efficiency, Transparency, and Innovation through Reusable and Open Source Software" defines proprietary software as "software with intellectual property rights that are retained exclusively by a rights holder (e.g., an individual or a company)."

Source Code means any sequence of computer programming statements or declarations written in human-readable computer or programming language.

Strongly Protective Open Source Licenses require that the original open source licensed software and derivative works based on the licensed software be distributed under the same terms as the original open source license. This prevents the open source software and any derivative works from becoming proprietary or being incorporated into any proprietary software. The GNU General Public License (GPL) is an example of a strongly protective open source license. In the open source community, "strongly protective" open source licenses are also known as "*strong* c*opyleft*" licenses. ***Strong Copyleft*** licenses require that

[2] Open Systems Architecture Contract Guidebook for Program Managers, Version 1.1, May 2013 https://www.dau.edu/cop/pm/DAU%20Sponsored%20Documents/Open%20System%20Architecture%20(OSA)%20Contract%20Guidebook%20for%20Program%20Managers-version%201.1-%20June%20%202013.pdf?Web=1.

derivative works be distributed under terms compatible with the original license. [3]

Unlimited Rights License means the license of the same name as defined at FAR 52.227-14, or the license of the same name as defined in an applicable agency-specific supplement to the FAR. The DFARS unlimited rights license is defined at DFARS 252.227-7014 (a) (15).

Weakly Protective Open Source Licenses allow some derivative works to be distributed under terms different than the original license, as long as the derivative work maintains sufficient separation from the original work, typically by interfacing with the original work as a distinct and separate component. This prevents the open source software component (often a software library) from becoming proprietary, yet permits it to be part of a larger proprietary program. Examples weakly protective open source license include the GNU Lesser General Public License (LGPL) and the Mozilla Public License. In the open source community, "weakly protective" open source licenses are also known as "*weak* c*opyleft*" licenses. ***Weak Copyleft*** licenses allow derivative works separate from the library itself to be distributed under terms different than the copyleft provisions of the original license.

[3] ***Copyleft*** is a general method for making a computer program or other work available free, and requiring all modified and extended versions of the program or work to be free as well, in an effort to include others in improving the program or as a continuing process. Copyleft licenses are referred to as "strong copyleft" or "weak copyleft," licenses depending on the extent to which they impose copyleft provisions on derivative works, words, numbers, or other verbal or numerical symbols or indicia, regardless of the nature of the material objects, such as books, periodicals, manuscripts, phono-records, film, tapes, disks, or cards, in which they are embodied.

2 Computer Software Copyright Basics

2.1 General Information Regarding Copyright and Computer Software

2.1.1 Is computer software subject to copyright protection under Section 102 of the Copyright Act?

Yes. Computer programs, which is the term used for computer software in the U.S. Copyright Act, are protected as "literary works" under Section 102 (a) (1) of the Act. *See e.g. Computer Associates, Inc. v. Altai, Inc*. 982 F. 2d 693 (2nd Cir. 1992). Literary works are "works" other than audiovisual works, expressed in words, numbers, or other verbal or numerical symbols or indicia. (See 17 USC §§ 101 and 102 (a) (1) and FAQ Section 2.1.3.[4]) Copyright protection in computer software extends to both the source code and the object code.

However, not all of the features of a computer program are copyrightable. For example, the First Circuit has ruled that a "menu command hierarchy" is considered a method of operation and uncopyrightable subject matter. (See *Lotus Development Corporation v. Borland International, Inc*) 49 F.3d 807 (1st Cir. 1995), aff'd, 516 U.S. 233, 116 S. Ct. 804, 133 L. Ed. 2d 610 (1996). In the Lotus case, the First Circuit reversed a district court decision holding that the Lotus menu command hierarchy was expressed in a particular way and was copyrightable. The Supreme Court affirmed the First Circuit's ruling without a decision. Therefore, the question of whether all menu command hierarchies are methods of operation and uncopyrightable remains an unsettled area of copyright law. "A "method of operation"[5] refers to the means by which a person operates something, whether it's a car, a food processor, or a computer. By definition (See Section 101 of the Copyright Act), computer programs are a set of statements or instructions that bring about a certain result, which is not very different from a method of operation. The question that the courts have not resolved is whether some methods of operation may include a particular expression that may be copyrightable.

More recently, issues concerning software and copyright have been at the core of intensive litigation between Oracle and Google. In November 2019, the Supreme Court granted certiorari in an effort to definitively resolve a nine-year battle between the parties concerning smartphone software, which is currently used in Google's Android mobile operating system (See Google LLC v. Oracle America, Inc., No 18-956). Prior to the Supreme Court taking the case, the Court of Appeals for the Federal Circuit (CAFC) had twice sided with Oracle on two key issues at stake in the case. First, the CAFC held that section 102(b) of the Copyright Act (which denies copyright protection for any idea, procedure, process, system, method of operation, etc.) does not preclude protection for otherwise original "declaring code" that forms part of Oracle's complex library of pre-written computer programs that app programmers use as "shortcuts" to build certain functions into their programs without the need to write the code

[4] 17 U.S.C. § 101; H. Rep. No. 94-1733, 94th Cong., 2d Sess. (Sept. 29, 1976); Computer Software Copyright Act of 1980, Act of Dec. 12, 1980, Pub L. 96-517, Sec. 10, 94 Stat. 3015; Atari Games Corp. v. Oman, 888 F. 2d 878 (D.C. Cir. 1989); Whelan Associates, Inc. v. Jaslow Dental Library, 797 F. 2d 1222 (3d Cir. 1986); Nimmer on Copyrights, Section 2.04[C].

[5] See 17 U.S.C. §102(b)

for these functions from scratch. See Oracle Am., Inc. v. Google Inc., 750 F.3d 1339 (Fed. Cir. 2014). On remand, a jury found that Google's copying of Oracle's declaring code was a fair use. On appeal, the CAFC reversed and held that fair use was not applicable because, inter alia, Google's copying was "overwhelmingly commercial," was not transformative in that Google used the copied code to perform the same functions as the original code, and inflicted actual and potential harm on the economic market for Oracle's work. See Oracle Am., Inc. v. Google, 886 F.3d 1179 (Fed. Cir. 2018).

Methods of operation, including methods implemented by software, may be protected by patent if the method satisfies the requirements for patentability. (See FAQ 2.1.3.2.)

For further discussion and information on registering computer programs with the U.S. Copyright Office, see Copyright Office Circular No. 61, Copyright Registration for Computer Programs, and the Compendium of U.S. Copyright Office Practices.

2.1.2 What rights are granted to owners of copyrights to computer software under Section 106 of the Copyright Act?

Owners of copyrights to computer software acquire the same exclusive rights as owners of any other literary work: the exclusive right to (a) reproduce the software; (b) prepare derivative works based upon the original software; (c) distribute the software; (d) publicly perform; and (e) publicly display the software. 17 U.S.C. § 106

Although copyright owners of computer programs generally license their software for use by others, they typically restrict a licensee's rights to modify, prepare derivative works, and distribute the computer program, and the owner(s) thereby retains these rights. Copyright owners commonly implement these restrictions by giving licensees access only to the object code and not the source code for the software. This is a key difference between a proprietary and an open source licensing model. Under open source licenses, copyright owners allow others to exercise their exclusive rights with few, if any, limitations by allowing users to modify the source code, and to prepare and distribute derivative applications, provided that if the modifications are distributed, the source code is shared with the community of users. (See FAQ 4.1 and 4.10.)

Because most computer programs are licensed and not sold, whether or not the licensee can sell or otherwise dispose of a particular "copy" of the licensed software is a separate question. For both section 109 (the "first sale" doctrine) and section 117 (limitation on exclusive rights in computer programs) of the Copyright Act, the answer will depend on the degree of ownership the licensee acquires in the transaction.

In a frequently cited case from the 9th Circuit, a licensee was not able to rely on section 109 as a defense to his sales of computer programs. *Vernor v. Autodesk, Inc.*, 621 F.3d 1102 (9th Cir. 2010). Autodesk, the computer program maker, had specified that it had granted users a non-transferable license and placed other restrictions on the use of the software. In reversing the district court's ruling for Mr. Vernor, the alleged infringer, the appellate court ruled that the alleged infringer was not an owner of a particular copy, and therefore could not make a non-infringing sale or transfer of the software. The court stated, in dicta, that the customers of the defendant-reseller could not rely on §117(a)(1) as a defense to any copies/reproductions of the copyrighted software that would have been made when said customer installed the software on their computers because said customers were, like the defendant-reseller, also not owners of the software copies.

In the reported decisions finding that a license precluded a section 109 or 117 defense, the computer programs involved were one-to-one transactions for relatively expensive software created for the particular end-user. Whether the defenses are precluded in the case of mass market software – where boilerplate terms of service agreements are commonly used (and seldom read by buyers) – is still an unanswered question.

2.1.3 Is it possible to protect computer software under other types of intellectual property law?

Yes, in addition to copyright protection, computer software may also be protectable under trademark[6], patent[7] and trade secret[8] law. More than one type of protection may apply to a single computer program.

2.1.4 May computer software be protected by trademark law?

Yes. A trademark, (any word, symbol, design, device, sound, logo, slogan or combination of these) that is used in connection with a computer program to identify its source (for example, "MS Word" and an image identifying Microsoft's word processing program) can be protected under trademark law, registered with the US Patent and Trademark Office and the trademark registries of other countries.

2.1.5 May computer software be protected by patent law?

Yes, under certain conditions. Protection of software with a patent is the result of a recent interpretation of the scope of patentable subject matter by the courts. In the late 1990s, the U.S. Patent and Trademark Office began issuing patents for software applications involving methods of operation or processes (aka "business method patents"), a practice which expanded rapidly following the Court of Appeals for the Federal Circuit's decision in *State Street Bank & Trust Co. v. Signature Financial Group Inc*., 149 F.3d 1368 (Fed. Cir. 1998). Business method patents have generated controversy in the U.S. and can be difficult and

[6] A trademark is a word, name, symbol, or device that is used in trade with goods to indicate the source of the goods and to distinguish them from the goods of others. A service mark is the same as a trademark except that it identifies and distinguishes the source of a service rather than a product. Trademark rights may be used to prevent others from using a confusingly similar mark, but not to prevent others from making the same goods or from selling the same goods or services under a clearly different mark.

[7] A patent for an invention is the grant of a property right to the inventor, issued by the United States Patent and Trademark Office. The right conferred by the patent grant is the right to exclude others from making, using, offering for sale, or selling the invention in the United States or importing the invention into the United States. What is granted is not the right to make, use, offer for sale, sell or import, but the right to exclude others from making, using, offering for sale, selling or importing the invention. Title 35, United States Code.

[8] Trade secret is information, including a formula, pattern, compilation, program, device, method, technique, or process that that has economic value by not being known to the public, and for which the owner takes reasonable steps to protect from disclosure. See generally *Congressional Research Service, Protection of Trade Secrets: Overview of Current Law and Legislation*, April 22, 2016.

expensive to obtain. Under a 2008 court decision, *In re Bilski*, 545 F. 3d 943, 88 U.S.P.Q 2d 1385 (Fed. Cir. 2008), the future of business method patents in the U.S. is unclear. The *Bilski* court ruled that the "useful, concrete, and tangible result" test used in *State Street* should no longer be relied upon. The court also reiterated the "machine-or-transformation test" as the applicable test for patent-eligible subject matter. Whether and to what extent business method patents (both new applications and patents already issued) may successfully meet this test remains to be seen.

The controversy surrounding the patentability of software applications remains unresolved after the U.S. Supreme Court's decision in *Bilski v. Kappos*, 561 U.S. 593, 130 S. Ct. 3218, 177 L. Ed. 2d 792 (2010) issued on June 28, 2010. The court affirmed the Federal Circuit court's decision invalidating *Bilski*'s patent. In addition, it also held that the "machine-or-transformation" test was not the exclusive test for determining whether a claimed process is patentable under Section 101 of Title 35. However, it avoided the larger question of the patentability of business methods software. Thus, the patentability of this type of software will continue to be reviewed on a case-by-case basis. Online resources for patent eligibility issues may be found at: https://www.uspto.gov/patent/laws-and-regulations/examination-policy/subject-matter-eligibility. Patent rights in software created under a government contract are generally addressed under FAR 52.227-11, 52.227-13 and 52.227-14. The DFARS addresses this issue at 252.227-7038.

Subject matter eligibility for software patents was addressed again in *Alice Corp. v. CLS Bank Int'l*, 573 U.S. 208, 134 S. Ct. 2347, 189 L. Ed. 2d 296 (Supreme Court 2014), June 19, 2014. The *Alice* court made use of the analysis in *Mayo Collaborative Services v. Prometheus Laboratories, Inc*., 132 S. Ct. 1289, 566 U.S. 66, 182 L. Ed. 2d 321 (2012), which set forth a two-step framework for distinguishing patents that claim patent-ineligible laws of nature, natural phenomena, and abstract ideas from those that claim patent-eligible applications of those same abstract ideas. The court quickly determined that the claimed process of intermediated settlement is an abstract idea. The case hinged on whether this unpatentable abstract idea was "transformed" into patentable subject matter. The court decided that by merely using a computer to implement the abstract idea does not create a patent-eligible invention.

However, *Alice* contrasts *Diamond v. Diehr*, 450 U.S. 175, 101 S. Ct. 1048, 67 L. Ed. 2d 155 (1981) which held that a computer-implemented process for curing rubber is patentable, not because it used a computer, but because the invention improved a technological process. The invention achieved constant temperature during the rubber curing process, something not previously achieved without the help of a computer.

Obtaining patent protection for computer software is a complex process and many practitioners prefer to protect their intellectual property through copyright. In January 2019, the US Patent and Trademark Office (USPTO) issued two notices: 2019 Revised Patent Subject Matter Eligibility Guidance [FR Doc. 2018–28282] and Examining Computer-Implemented Functional Claim Limitations for Compliance with 35 U.S.C. 112 [FR Doc. 2018–28283]. More specifically, the Patent Office's current eligibility guidance is found in the 2019 Revised Patent Subject Matter Eligibility Guidance (2019 PEG), October 2019 Patent

Eligibility Guidance Update (October 2019 Update), and in Manual of Patent Examination Procedure (MPEP) Sections 2103, 2104, 2105, 2106 and 2106.03 through 2106.07(c) (except 2106.04(II) which is now superseded). (See Additional information on subject matter eligibility on the USPTO website at https://www.uspto.gov/patent/laws-and-regulations/examination-policy/subject-matter-eligibility.)

2.1.6 May computer software be protected as a trade secret?

Yes, computer programs may be protected as trade secrets under both state and federal law and various licensing arrangements. While trade secrets law developed in the U.S. through the common law among the various States, a large majority of States have now adopted some variant of the Uniform Trade Secrets Act (UTSA). The UTSA defines "trade secret" to mean information, including a formula, pattern, compilation, program device, method, technique, or process, that:

(i) derives independent economic value, actual or potential, from not being generally known to, and not being readily ascertainable by proper means by, other persons who can obtain economic value from its disclosure or use, and

(ii) is the subject of efforts that are reasonable under the circumstances to maintain its secrecy.

Moreover, the Defend Trade Secrets Act of 2016 (DTSA) (Pub.L. 114–153, 130 Stat. 376, enacted May 11, 2016, codified at 18 U.S.C. § 1836, et seq.) is a United States federal law that allows an owner of a trade secret to sue in federal court when its trade secrets have been misappropriated. NOTE: DTSA does not create a private right of action in regard to any otherwise lawful activity conducted by a governmental entity of the United States, a State, or a political subdivision of a State. (18 U.S.C. § 1833(a)(1).)

There is also a federal criminal statute providing for prosecution of theft of trade secrets, which contains a lengthier, but similar, definition. (See 18 USC 1839, et seq.) (There is also another federal criminal statute, 18 USC 1905, commonly referred to as the "Trade Secrets Act," but this statute does not contain a definition of "trade secret" and primarily addresses the confidentiality obligations of federal employees in performance of their official duties.)

Computer program trade secret claimants often employ protective measures such as licensing agreements containing confidentiality provisos, non-disclosure agreements for third-party code developers, distribution of the software only in executable form, and physical security for source code copies. The U.S. Copyright Office recognizes trade secret claims in computer programs and provides several registration options for the deposit of only a portion of the code. (See U.S. Copyright Office Circular No. 61, Registration for Computer Programs.)

3 Open Source Software (OSS): The Basics

3.1 What is OSS and how does it differ from proprietary software?

OSS is software distributed under a license that typically provides broad rights to use, modify, and redistribute the original source code and, oftentimes requires open distribution of, any modified versions as well (i.e., derivative works). Many different OSS license types exist; each imposes certain obligations that have various legal implications. For example, many open source licenses do not require the payment of royalties or mandate redistribution limits typically associated with proprietary software licenses. Most open source licenses automatically terminate if the licensee violates the license. Thus, once a licensee violates the terms of an open source license, the licensee has breached the contract and has infringed any copyright in the OSS.

Most open source licenses impose a share-alike clause that requires any redistribution of the original open source code and any derived works to be under the same or similar open terms as the original license. Open source licenses ensure that the rights granted cannot later be revoked and that derivative works must be provided in a form that facilitates modification. For software, this requires that the source code of the derivative work be made available with the software itself. Open source proponents claim that the share-alike clause fosters the free and open development and improvement of the software by a broad OSS development community and equal participation by all users, while opponents claim that share-alike creates undesirable licensing complications and restrictions.

Open source licenses are typically referred to as being “strongly protective” (aka “strongly copyleft”) or “weakly protective” (aka “weakly copyleft”), based on the extent to which open source provisions can be imposed on derived works. Strongly protective licenses require that any derivative works be distributed under a compatible open source license. Thus, strongly protective licenses are sometimes referred to as viral licenses because any software incorporates or derives from the licensed work (even if that software is originally proprietary code) must be released under a compatible open source license.

Weak protective licenses require derivative works of the OSS to be redistributed under the same or similar open source terms, but specifically allow other software to link to the original OSS or derivative work without imposing the open source license requirements on the related software. Only changes to the OSS itself become subject to the open source provisions, not the software that uses it through a standard interface. This allows software distributed under other licenses (including proprietary licenses) to be linked to the weakly protected software, and then be redistributed under its own terms.

Permissive licenses do not impose the share-alike clause. Permissive licenses place limited restrictions such as crediting the original author and stating that the original author makes no warranties on the work. Permissive licenses permit redistribution of the original and derivative works of the OSS under their own terms and conditions, which can differ from those of the original work. Therefore, permissive licenses offer many of the same freedoms as releasing a work to the public domain. Thus, derivative works can become proprietary and the original OSS can be incorporated into proprietary software.

3.2 What are some of the common open source licenses and their distribution terms?

There are many open source licenses in use. The Open Source Initiative (OSI), a prominent non-profit that promotes OSS, recognizes over 80 licenses that it considers to be open source[9]. Open source licenses certified by OSI must satisfy the Open Source Definition (http://www.opensource.org/docs/osd) that requires open source licenses to meet the following distribution terms:

(a) The software must be freely distributed;
(b) The software must be distributed in source code as well as compiled form and a well-publicized means for obtaining the source code;
(c) The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software;
(d) Software built from modified source code must be distributed;
(e) The licensing terms must not discriminate against persons or groups of persons;
(f) The license must not restrict anyone from using the program in a specific field or endeavor;
(g) The licensing rights must apply to all to whom the program is redistributed without the execution of an additional license;
(h) The licensing rights attached to the program must not depend on the program's being part of a specific program;
(i) The license must not place restrictions on other software distributed along with the program; and
(j) The license must be technology neutral.

The Linux Operating System is distributed under the GNU GPL license which is an example of a strongly protective license. Examples of weakly protective licenses include the GNU (LGPL) and the Mozilla Public License. Popular, permissive OSS licenses include Apache licenses (all except v1.0), the BSD (Berkeley Software Distribution) License and the MIT (Massachusetts Institute of Technology) License. Some federal agencies, such as NASA, have created their own licenses. (See Section 4.3.) The Open Source Initiative (OSI) has certified the NASA Open Source Agreement (NOSA).

Creative Commons (CC) does not recommend its licenses for OSS. The CC FAQ notes "CC licenses may be applied to any type of work, including educational resources, music, photographs, databases, government and public sector information, and many other types of material. The only categories of works for which CC does not recommend its licenses are computer software and hardware." CC licenses do not address distribution of source code; and they are not wholly compatible with the most frequently used licenses (which makes integration with software covered by other open source licenses problematic). CC-0[10] may be used to dedicate copyright and related rights to the public domain, however the CC-0 dedication has not been approved by the Open Source initiative.

License Compatibility: If two or more programs, covered by different OSS licenses will be

[9] See https://opensource.org/licenses/alphabetical
[10] See https://wiki.creativecommons.org/wiki/CC0_FAQ

combined into a single larger work, a legal analysis is needed to determine if it is possible to comply with all of the conditions set forth by the individual licenses. The widely-used licenses tend to be compatible, i.e., the software can be combined to produce a larger work. The chart below illustrates in a general manner, the compatibility between permissive and protective licenses. An arrow between two licenses indicates that they are typically compatible. However, each license should be examined carefully to ensure that both or all of the licenses can be used simultaneously.

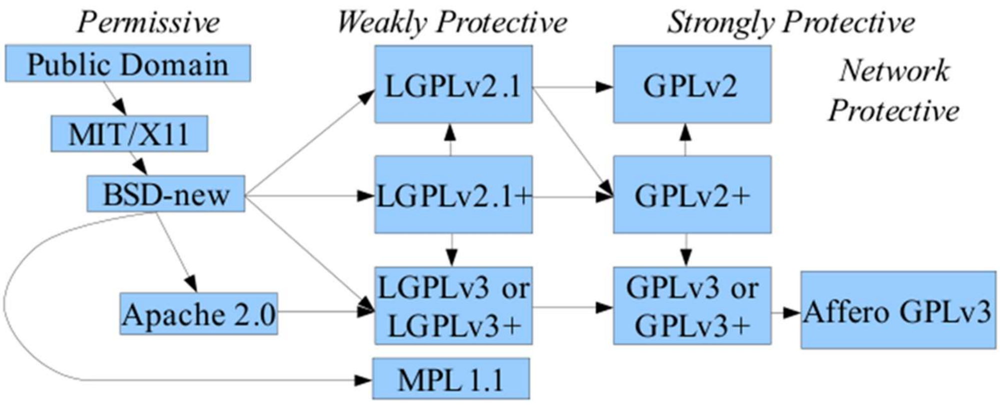

Chart reproduced from https://dwheeler.com/essays/floss-license-slide.html under a Creative Commons [BY-SA 3.0 US](#) license.

Other resources on license compatibility include:

GNU License compatibility at https://www.gnu.org/licenses/gpl-faq.html#AllCompatibility

The Free Software Foundation “Various Licenses and Comments about Them” at https://www.gnu.org/licenses/gpl-faq.html#AllCompatibility

Eckert, Lauren A. U.S. Army Engineer Research and Development Center (ERDC). “Open Source Software Compliance within the Government” ERDC/ITL SR-16-32 available at https://apps.dtic.mil/dtic/tr/fulltext/u2/1027801.pdf

3.3 How does an open source licensing model impact the exclusive rights granted to copyright owners under Section 106 of the Copyright Act?

Section 106 of the Copyright Act provides copyright owners with five exclusive rights: (1) the right to reproduce; (2) the right to prepare derivative works based upon the original copyright work; (3) the right to distribute the copyrighted work; (4) the right to publicly perform; and (5) the right to publicly display the copyrighted work.

Copyright owners’ licensing software using a proprietary license typically restrict a licensee’s rights to modify, prepare derivative works, and distribute the software program. By allowing licensees to freely modify, create derivative works, and redistribute the original and modified source code, the open source copyright owner is providing a broad,

This document does not constitute legal advice and should not be construed or used as such.

but revocable, license of its exclusive rights. If the licensee violates the terms of the open source license, the open source licensor can terminate the license (actually, most open source licenses automatically terminate upon violation of their terms) or enforce the license under both copyright and contract law.

The open source copyright owner(s) may provide the broad license of his/her rights for a variety of motivations (e.g., to encourage others to contribute improvements, to gain recognition/reputation, to gain a competitive advantage, for which the company may sell service/support, or simply to provide a service to the world).

4 Computer Software and the U.S. Government

4.1 Have U.S. Government agencies issued policy guidance regarding the use of OSS?

Yes. OMB issued a new OSS policy, "Federal Source Code Policy: Achieving Efficiency, Transparency, and Innovation through Reusable and Open Source Software" (M-16-21) in 2016 requiring federal agencies to release at least 20 percent of new custom-developed source code as OSS or without any restrictions on use. The policy promotes government-wide reuse of custom-developed code created by agency employees or contractors on behalf of the government when it obtains sufficient data rights to redistribute the code. Ensuring government-wide reuse rights for federal custom-code should help eliminate duplicative acquisitions of "substantially" similar code and inefficient use of taxpayer dollars. To realize these benefits, the Policy instructs agencies to comply with the following requirements:

(a) secure rights for government reuse and ensure delivery of source code in contract negotiations;
(b) inventory all custom-developed code and make it available government-wide; and
(c) as part of a 3-year pilot program, release at least 20 percent new custom-developed code potentially useful to the broader community.

Consistent, agency-specific guidance concerning use of OSS may also exist.

4.1.1 Agency implementation: Department of Defense

The DoD CIO maintains a webpage on OSS as "an educational resource for government employees and government contractors to assist in understanding the policies and legal issues relating to the use of OSS in the Department of Defense (DoD)" that includes Open Source Software FAQ and Clarifying Guidance Regarding Open Source Software. In 2007, the U.S. Department of the Navy (DoN) issued a policy memo recognizing OSS as commercial-off-the-shelf (COTS) when it meets the definition of a commercial item pursuant to Section 403 of Title 41, and encouraging its use in IT acquisitions when it complies with Federal, DoD, and DoN policies. The U.S. Army issued a regulation, US Army Regulation 25-2 Information Assurance, that at paragraph 4-6.h, provides guidance on software security controls that specifically addresses OSS.

4.1.2 Agency Implementation: Department of Energy

The Department of Energy (DOE) Office of Scientific and Technical Information (OSTI) developed DOE CODE, a software service platform and search tool that allows for scientific and business software to be provided to DOE. DOE CODE provides functionality for collaboration, archiving, and discovery of scientific and business software. In addition to OSTI's charge for scientific software, the DOE Office of the Chief Information Officer (OCIO) and OSTI partnered to leverage DOE CODE to maintain a comprehensive inventory of DOE-funded custom-developed business software. In accordance with Office of Management and Budget Memorandum M-16-21, Federal Source Code Policy: Achieving Efficiency, Transparency, and Innovation through Reusable and Open Source Software, DOE CODE fulfills the requirements to maintain an inventory of all DOE-funded custom-developed software and to report this inventory to the government-wide Code.gov website.

4.1.3 DoD Open Systems Architecture (OSA)

In June of 2013, DoD's OSA Data Rights Team Contract published a comprehensive guidebook, "DoD Open Systems Architecture Contract Guidebook for Program Managers: A Tool for Effective Competition" targeted at Program Managers involved in the acquisition of technology using modular designs based on standards with loose coupling and high cohesion that allow for independent acquisition of system components. Appendix 5: Open Source Software of this guidebook includes specific guidance on the distinction between OSA and OSS and may be useful for Program Managers and Contracting Officers.

4.1.4 Agency implementation, Civilian Agencies

Some civilian agencies such as NASA, release OSS under agreements that are certified as open source licenses by the Open Source Initiative (OSI).[11]

Similarly, the National Library of Medicine at the National Institutes of Health released its Strategic Plan 2017-2027, committing to the creation of new methodologies and new ways of organizing collections of data science tools, including OSS and algorithms. More broadly, the NIH Strategic Plan for Data Science seeks to promote "community-guided development of model open data-use licenses that will facilitate data sharing while simultaneously allowing protection of confidentiality and intellectual property."

The Environmental Protection Agency (EPA) issued interim policy to implement the requirements of the OMB Source Code Policy. The policy applies to new custom-developed code created or procured by EPA.

4.2 Where have U.S. Government agencies made their OSS available?

18F is an office of federal employees within the General Services Administration (GSA) that collaborates with other agencies to fix technical problems, build products, and improve how government serves the public through technology. 18F is part of the Technology Transformation Services, which is within the Federal Acquisition Service maintains a webpage

[11] NASA Open Source Agreement (NOSA). See "NASA Open Source Agreement" and "NASA Procedural Requirements 2210. An External Release of NASA Software."

that contains facts and references about publishing OSS in public code repositories. In fact, many agencies and offices of the federal government use public hosting services including Agriculture, Commerce, Education, Homeland Security to name a few. See the comprehensive federal GitHub dashboard created by GSA's Office of Government-wide Policy.

Examples of specific agency projects include:

4.2.1 Code.gov

Code.gov is a centralized resource for information on open source policy, procurement best practices, software inventory procedures and compliance. There is also an interface to assist developers as well as a browse feature with access to over 5,000 federal open source projects.

4.2.2 Department of Defense (DoD)

In 2017, to further the mission of "Code.gov" and follow the policy guidance in the Federal Source Code Policy and address DoD's unique challenges in using OSS, the Defense Digital Service (DSS), a DoD agency team of the U.S. Digital Service, launched "Code.mil." Code.mil is an experiment in open source designed to foster open collaboration with the developer community across the world on DoD open source projects. The platform was launched in three phases: (1) developer community provided input on DoD open source strategy; (2) launched first open source projects in March, 2017; and (3) sustainment by adding additional projects from DoD offices, tackling procured code and discussing how to facilitate technology transfer of open source code. See https://code.mil.

4.2.2.1 Army Research Lab (ARL)

U.S. Army Research Laboratory (ARL) Software Release Process for Unrestricted Public Release Version 1.0.4 28 July 2017, provides procedures that ARL government personnel must follow when releasing software source code and software-related material to the public, and for accepting software-related contributions from the general public.

4.2.2.2 Defense Information Systems Agency (DISA)

DISA merged the concepts of technology transfer and licensing government-created software via an OSS platform in 2009 by entering into a Cooperative Research and Development Agreement (CRADA) with the Open Source Software Institute (OSSI). The CRADA facilitated the release of DISA's internally-developed Corporate Management Information System (CMIS) application suite via an open source licensing scheme for the purpose of creating a collaborative partnership with other government agencies, industry and academia to research and develop enhanced functionality for DISA software that is for the use of DoD, the federal government, state and local governments and the public.

4.2.3 Department of Energy

In 2017, the Department of Energy (DOE) Office of Scientific and Technical Information

(OSTI) developed a new DOE software services platform and search tool for DOE-funded code – DOE CODE. DOE CODE replaced the centralized software management facility for DOE, the Energy Science and Technology Software Center (ESTSC), launched in 1991. DOE CODE is an open source platform that makes it easy for DOE-funded researchers and scientific software developers to submit scientific software to DOE and to obtain digital object identifiers (DOIs) for software projects, thereby enabling the public to discover and use DOE-funded code. It fulfills Departmental requirements to maintain an inventory of all DOE-funded custom-developed software and is used to report this inventory to the government-wide Code.gov website.

4.2.4 National Aeronautics and Space Administration

NASA launched its code directory code.nasa.gov in January 2012 and continues to publish open source projects through this portal. The website will continue to unify and expand NASA's open source activities, serving to surface existing activities, provide a forum for discussing efforts and processes, and guide internal and external groups in open development, release, and contribution. NASA uses multiple public, open source development repositories at SourceForge and GitHub to host NASA OSS releases.

4.3 How do the FAR and DFARS address the use of OSS?

Both the FAR and DFARS treat OSS as "commercial software," which would be licensed to the government under the same terms as licensed to the general public. 41 USC § 403 defines a commercial item for purposes of both the FAR and DFARS as: "[ ] any item, other than real property, that is of a type customarily used by the general public or by non-governmental entities for purposes other than governmental purposes, and (i) has been sold, leased, or licensed to the general public or (ii) has been offered for sale, lease, or licensed to the general public."

DFARS 227.7202-1(a) provides that commercial computer software shall be acquired under licenses customarily provided to the public unless such licenses are inconsistent with federal procurement law or do not otherwise satisfy the government's needs. FAR section 12.212 provides that commercial software is acquired under licenses customarily provided to the public to the extent such licenses are consistent with federal law and otherwise satisfy the government's needs. Because OSS is licensed to the public and not developed exclusively for government use, it meets the definition of commercial software and would be licensed to the government under the same open source terms as to the general public.

In cases where commercial software, including OSS, is used as part of an application created by a contractor for government use, as noted in Section 4.4 of the "Frequently Asked Questions About Copyright," the contractor should seek the government's permission to use the OSS. The contractor should also provide a copy of the license to the agency for review by Intellectual Property (IP) counsel to ensure that the terms of use do not pose problems, e.g. the terms of use are not consistent with federal procurement law or do not otherwise satisfy the government's needs as detailed in DFARS 227.7202-1(a), and FAR 12.212

4.4 Are there issues unique to federal agencies in distributing OSS?

Yes, a civilian or military agency may distribute OSS if they have sufficient ownership interests or licensing rights in the software. For example, agencies typically use and may want to distribute to other users, within and outside the government, software created by: (1) its employees as part of their official duties; (2) a vendor, acting on the agency's behalf within the context of a procurement or other award instrument; and (3) a vendor who licenses its software using an open source licensing scheme. An agency seeking to distribute OSS developed under any of these scenarios must first decide whether it owns or has acquired sufficient licensing rights to make the software available to other users.

With the exception provided under changes to 17 USC § 105, which allows a civilian member of the faculty of certain listed U.S. Government institutions (see PL 116-92 §544) to own copyright to certain works, copyright protection is not available in the U.S. for software created by government employees as part of their official duties (See 17 USC § 105). However, copyright ownership is not a necessary prerequisite to adopting an open source strategy. An open source license is a contract, and even if the agency does not own the copyright for the code, the agency may still be able to obtain a license to distribute the code for agency purposes.

For example, an agency may distribute software created by a vendor to users under an open source licensing scheme if the agency acquired sufficient rights in the software from the vendor. One method for acquiring such rights is defined in both the FAR and DFARS, in which an agency funds, wholly or partially, the creation of software. Where the agency funds the creation of the software the agency generally obtains an "unlimited rights license". Where the agency jointly funds the development of the software with the vendor, it generally acquires a "government purpose rights license". Under either license type, an agency may be able to distribute the software to third parties under an open source-model, as long as it complies with any restrictions attached to the software under the original contract, i.e., the license obtained. Agencies wishing to disseminate OSS or participate in open source development may wish to include explicit open source requirements in their contracts and grants for software development.

Federal agencies may also wish to distribute applications—created by their employees or vendors, acting on their behalf—which include OSS components. Prior to choosing this option, agencies must carefully evaluate their own licensing rights under the original contract or other award instrument, as well as the requirements of the particular open source licensing scheme under consideration.

Given the complex and often confusing issues posed by software acquisitions of all types and the use and licensing of proprietary and OSS, program managers should always consult their agency's acquisition and IP counsel and contracting officers prior to sharing or disclosing software to any government or other user.

Finally, where it has sufficient ownership or licensing rights to do so, a federal agency may wish to coordinate use of OSS licenses and the distribution of the software through already existing open source portals such as SourceForge. DoD agencies also have the option of distributing software through "Forge.mil," a web site enabling the collaborative development and distribution of OSS and DoD community source software. Some civilian agencies also distribute their OSS through agency specific web sites. NASA, for example, distributes its OSS through both their NASA.gov web site (see https://code.nasa.gov/) and through an agreement with SourceForge. Similarly, the Department of Transportation has established ITSForge.net, a collaborative open source application development site for the creation and public distribution of OSS relating to intelligent transportation systems.

4.5 Is the U.S. Government allowed to use OSS on government computer networks?

Generally yes, when it best fits the needs and mission requirements of the agency involved and it meets applicable information assurance or other security standards for the particular computer network on which it is to be used. Many types of OSS are widely used in the U.S. Government, for example the Drupal content management system (see Content Management Systems Used by Government Agencies), Linux kernel, Samba, Apache, Perl, GCC, GNAT Ada Compiler and others. The U.S. Government has also released both entirely new programs and improvements of existing OSS (e.g., OpenVista, Expect, Security Enhanced Linux). As noted in Memorandum M-16-21, Federal Source Code Policy: Achieving, Efficiency, Transparency and Innovation Through Reusable and Open Source Software (August 8, 2016), a significant proportion of software used by the government is comprised of either preexisting Federal solutions or commercial solutions. These solutions include proprietary, open source, and mixed source code.

4.6 Are there any special issues involved in government use of OSS?

The same issues should be considered for OSS as for proprietary software. These issues can be roughly categorized into three main groups, and each of them should be fully assessed prior to any software purchase or use, whether OSS or proprietary: (1) copyright status, license and contractual terms, (2) acquisition life cycle, and (3) cybersecurity.

4.6.1 What are the OSS copyright licensing and contractual considerations of greatest concern to the government?

All copyright licensing and contractual terms should be carefully assessed to ensure that the government can legally agree to them and fully understands the risks involved in accepting them, particularly provisions addressing warranties, indemnifications, distribution and redistribution of code, patent licenses, and applicable law and dispute resolution mechanisms.

4.6.2 What are the main OSS acquisition life cycle considerations?

All aspects of the acquisition life cycle should also be fully analyzed, including determining the “total cost of ownership” of the software. Low initial purchase price is often a very attractive feature of OSS, but many other costs should also be carefully considered.

Characteristics of the software itself should be assessed, including its accountability, integrity, reliability, scalability, and flexibility. Transition costs include software configuration and installation, file backups, data file format conversions, and new hardware installation. Training costs include training for help desks and administrators as well as users. Maintenance costs include onsite maintenance and code tracking, as well as patching, adding new functional requirements, etc. Additionally, a potential user should vet the current market share and growth path of the particular OSS before selecting for government use. If continuous public maintenance and upgrade of the particular OSS under consideration is questionable, additional government resources may need to be allocated to replace dwindling public resources.

4.6.3 What are the main OSS security assessment considerations?

Proper security assessment for any software, whether OSS or proprietary, is extremely complex and requires special technical expertise. Technical and cybersecurity personnel should lead this assessment, and acquisition and legal counsel should work closely with them before any decisions are implemented. Among other things, security from the government perspective involves compliance with the intergovernmental Committee on National Security Systems (CNSS) Policy 11 - Acquisition of Information Assurance (IA) and IA-Enabled Information Technology (IT) Products and the Federal Information Security Modernization Act of 2014 (FISMA 2014) which amended the Federal Information Security Management Act of 2002 (FISMA) and updated the federal government’s cybersecurity practices as well as meeting applicable agency configuration and information assurance guidelines.

The Federal Risk and Authorization Management Program (FedRAMP) is a government-wide program managed by government and industry cybersecurity and cloud experts that provides a standardized approach to security assessment, authorization, and continuous monitoring for cloud products and services.

There are programs designed to address security on certain contractor information systems. Contractor information systems that process, store, or transmit information provided by or generated for the government under a contract must implement certain measures and controls to protect information not intended for public release. For civilian agency contractors, these requirements are listed at FAR 52.204-21 – “Basic Safeguarding of Covered Contractor Information Systems.” Defense contractors must implement security requirements in National Institute of Standards and Technology (NIST) Special Publication (SP) 800-171, “Protecting Controlled Unclassified Information in Nonfederal Information Systems and Organizations.” Furthermore, contractors must report any cyber incident to the government within 72 hours of discovery and preserve and protect images of all known affected information systems. DFARs 252.204.7012 “Safeguarding Covered Defense Information and Cyber Incident Reporting.”

4.7 Are there any particular advantages to government use of OSS as compared to using proprietary software?

Regardless of whether a particular piece of software is OSS or proprietary, the government should carefully assess its advantages and disadvantages—within the specific context of its intended use on an identified computer network—by examining each of the three issue categories identified in FAQ 4.5.

OSS, however, may have certain inherent advantages for the government that should also be factored into any acquisition or use determination. First, well-established OSS products may be inherently more reliable and more secure than proprietary products available for similar use. This is because OSS is often developed via a public, community-based approach, so it is also continuously subjected to very broad peer review and user assessment. Since the reviewing/user community is generally much wider for OSS than for comparable proprietary software, defects and vulnerabilities in the software may be identified earlier and fixed sooner than is possible with most proprietary products.

Other advantages to the government that are inherent in OSS include:

(1) access to source code, allowing government modification to fit particularized needs, rapid response when needs change or new threats are identified, and in-depth security review and audit;
(2) reduced dependence on a particular vendor, developer, or product, since OSS can be operated and maintained by multiple entities and many OSS products are easily interoperable with others;
(3) potential cost savings resulting from no "per seat" or "per copy" or field-of-use licensing fees and shared (community-based) maintenance/support costs; and
(4) greater adherence to the statutory preference for acquisition of commercial items over noncommercial items.

Foreign governments may view reduced dependence on specifically identifiable foreign suppliers (i.e. U.S. or European software sources) as an additional attractive feature of OSS. Some of the same advantages, such as greater adherence to the statutory commercial item preference, may also exist (or at least be negotiable, even if at increased price) for proprietary software.

4.8 Are there any particular disadvantages to government use of OSS as compared to using proprietary software?

Before acquiring or using any software, the government should ensure that the terms of the applicable license are compatible with the government's intended use, users, and identified computer network on which the software will be run. Although open source licenses, particularly the GPL, are popular and widely used by the open source community, a mandatory source code distribution requirement may not be appropriate for all government uses. The GPL only requires source code distribution when 1) the software is modified or 2) the software code is distributed to others; just using unmodified GPL software internally does not trigger the source code distribution clauses. While many government lawyers believe that no public distribution or propagation of modified code occurs so long as the code is used only within the federal government (including federal support contractors operating under nondisclosure agreements), this interpretation has not been reviewed by any court. Particularly where software potentially subject to a mandatory distribution licensing provision will be modified for use on, or linked to, classified or other secure computer systems, or where such software is export-controlled in accordance with the Arms Export Control Act,[12] and relevant agency regulations government managers should include these considerations as part of their risk assessment. Mandatory distribution requirements may also adversely implicate third-party proprietary code or information, depending on computer system and software architecture. Thus, government acquisition planners should ensure they consult closely with appropriate technical and legal advisors and fully understand the effects of such licensing provisions in advance.

Some OSS licenses also contain patent clauses which prohibit distributors from including code which requires a patent license to use, unless a royalty-free patent license is provided to all downstream users of the software and any derivatives. Some also have prohibitions against engaging in patent litigation related to the OSS, with the OSS license terminating for parties engaging in such patent litigation. The ramifications of these types of provisions must also be well understood beforehand.

Code covered by patents could effectively lose its patent protection if mingled with some types of OSS, particularly those with patent clauses which require royalty-free patent licenses. Such clauses are usually found in copyleft licenses like GPL, and not in permissive licenses like BSD or MIT. Further, OSS publication requirements may preclude the later integration of OSS based code with other proprietary software development, depending on the OSS license used. This can significantly delay the development of integrated tools, especially with contractor developed proprietary software.

Although many OSS products exist and many entities are available that provide OSS support and maintenance services, some OSS may not have a large enough supporting developer/user community to ensure that sufficient public maintenance and support of the software will remain available during the government's foreseeable use period. In such cases, the government should assess (prior to acquisition or use) whether it can allocate sufficient labor resources for these purposes from its own employee or contractor communities and can justify any additional costs this might entail.

Note also that special licensing terms deviating from the applicable OSS license generally cannot be negotiated by the government, since one or more of the relevant copyright holders may not be available to consent to altered licensing terms. An exception to this general rule is some or all of the OSS licensed by the Free Software Foundation (FSF); since the same organization is the sole copyright holder of all FSF software, its representatives may be able to negotiate special licenses in appropriate cases.

4.9 Is the U.S. Government required to give preference to proprietary software over OSS, or vice versa, in its acquisitions?

[12] DoD CIO Memorandum, "Clarifying Guidance Regarding Open Source Software (OSS)," October 16, 2009. The memorandum addressed the effective use of OSS within the Department and clarified implications of existing policy and regulation. The memo advised that some open source licenses permit the user to modify OSS for internal use without being obligated to distribute source code to the public and therefore may be integrated or modified for use in classified or other sensitive DoD systems. https://dodcio.defense.gov/Portals/0/Documents/OSSFAQ/2009OSS.pdf

There is no requirement to give preference to either proprietary or OSS over the other. However, the U.S. Government is required to give preference to commercial items over noncommercial items in its acquisitions, in accordance with 41 USC 3307. Nearly all OSS is commercial software, as are many proprietary software products.

5 Case Law on OSS Licensing: U.S. and International

5.1 Is there any U.S. federal case law addressing OSS licensing?

The Court of Appeals for the Federal Circuit (CAFC) squarely considered the issue of the enforceability of open source license in *Jacobsen v. Katzer*, 535 F.3d 1373 (Fed. Cir. 2008). Applying the interpretive law of the 9th Circuit, the Court found the terms of the Artistic License were enforceable copyright conditions, potentially allowing for injunctive relief against infringement, rather than merely contractual covenants remediable only by monetary damages. Citing an 11th Circuit opinion from 2001 (*Planetary Motion v. Techplosion*, 261 F.3d 1188 (11th Cir. 2001), the Court explained that substantial economic benefits can accrue to copyright holders under open source licensing, despite the fact that traditional copyright royalties are not generated. For example, the Court noted, economic benefits of open source licensing include allowing program creators to generate program market share by providing some components without charge, increasing professional reputation through open source project incubation, and obtaining rapid, free, and expert product improvements. The Court found that the clear restrictions contained in the license, including the requirement to retain reference to the original source files in modified or distributed code, were necessary to accomplish the objectives of the open source collaboration, and might well be rendered meaningless without the ability to enforce them through injunctive relief.

Although there have been a number of other federal lawsuits filed alleging infringement of open source licenses, most of them have been settled prior to judgment, and, apart from *Jacobsen,* none have yet resulted in substantive judicial interpretation of any open source license.

In particular, the enforceability of the GNU GPL, although raised in a number of disputes since 2008, has not yet been ruled on by a U.S. Federal Court. The issue has been the subject of a complaint filed in the Southern District of New York and contested in a number of settled U.S. cases. In the Southern District of New York, the Free Software Foundation filed a copyright infringement lawsuit against Cisco Systems, Inc., alleging violation of three GNU open source licenses, including the GPL, for OSS used in hardware devices sold commercially to the public. (*Free Software Foundation, Inc., v. Cisco Systems, Inc., 08-cv-10764, S.D.N.Y., complaint filed December 11, 2008.)* Free Software Foundation, Inc., v. Cisco Systems, Inc., involved an alleged violation of copyrighted code owned by the Freedom Software Foundation (FSF) in programs licensed under the GNU General Public License and the GNU Lesser General Public License. FSF claimed that products Cisco sold under the Linksys brand violated the licensing terms of its copyrighted programs, including GCC, GNU Binutils, and the GNU C Library. FSF's complaint asked the court to grant injunctive relief and enjoin Cisco from further distributing the Linksys firmware containing its copyrighted code. The parties settled the case with Cisco agreeing to appoint a director to ensure compliance with the open source licenses and making a financial contribution to FSF. [13]

Other settled cases include the "BusyBox" litigation in which the Software Freedom Law Center (SFLC), "filed a series of copyright infringement lawsuits" in which it sought to enforce version 2 of the GNUGPL (GPL v2) against various companies that had included GPL-licensed Busybox software into products offered for commercial sale, without releasing modified code back to the public.

> *Andersen v. Monsoon Multimedia, Inc.,* 07-cv-08205-JES, S.D.N.Y., complaint filed September 19, 2007*; Andersen v. High Gain Antennas, LLC,* 07-cv-10456, S.D.N.Y., complaint filed November 19, 2007*; Andersen v. Xterasys Corporation,* 07-cv-10455, S.D.N.Y., complaint filed November 19, 2007*; Andersen v. Verizon Communications Inc.,* 07- cv-11070, S.D.N.Y., complaint filed December 6, 2007*; Andersen v. Bell Microproducts, Inc.,* 08-cv-5270, S.D.N.Y., complaint filed June 9, 2008*; Andersen v. Super Micro Computer, Inc.,* 08-cv-5269, S.D.N.Y., complaint filed June 9, 2008.

Two other settled cases concerning OSS are *Progress Software Corp. v. MySQL AB,* 195 F. Supp. 2d 328 (D. Mass. 2002). and MySQL v NuSphere dispute.

In one well-publicized case, the SCO Group sued Novell, claiming IBM had infringed SCO's Unix-related copyrights by allowing copyrighted code to be released into the public domain in support of a Linux open source project, but the court held that since SCO did not own the copyrights, it lacked standing to sue for copyright infringement. *(SCO Group, Inc. v. Novell, Inc.,* 377 F. Supp. 2d 1145 (D. Utah 2005).

In another case, a software developer unsuccessfully alleged that IBM, Red Hat, and Novell used the GPL to fix software prices (at $0) in a pooling and cross-licensing scheme illegal under antitrust law that prevented the software developer plaintiff from competitively marketing his own software. (*Wallace v. International Business Machines Corp.,* 467 F.3d 1104 (7th Cir. 2006).

A recent case filed in 2017 in a New York federal court, CoKinetic Systems Corporation v. Panasonics Avionics Corporation, (1:17-cv-01527) District Court, S.D. New York, also settled prior to reaching trial. CoKinetic, seeking $100 million in damages, alleged that Panasonic willfully violated GPL v2 licensing requirements by failing to distribute the source code for its operating system, which is a Linux-based system licensed under the GNU GPL requiring distribution to third parties. CoKinetic argued that Panasonic's goal was to stifle competition in the in-flight entertainment industry. The parties ultimately settled the case.

The Versata Software case, considered a "ground breaking decision" in OSS because it raised the question (although failing to settle it) of whether software companies can restrict their contractors from redistributing their GPLv2 license. The lawsuit was filed in a Texas state court on May 3, 2013 and alleged that Ameriprise Financial Inc. materially breached a software license between the two parties for Versata's Distribution Channel Management (DCM) software, which Versata had licensed for millions of dollars. Versata was permitted to use "VTD-XML" software, an open source product owned by XimpleWare. Pursuant to

[13] PC World, Cisco Settles Lawsuit with Free Software Foundation", May 20, 2009.

Master License Agreement between Versata and Ameriprise, Ameriprise, its employees and certain contractors were allowed to use Versata's DCM software, which included the "VTD-XML" product. However, Ameriprise permitted non-permitted contractors to access and work on DCM, leading Versata to claim that it had infringed the MLA. Ameriprise counterclaimed that under the terms of the GPL, Versata was required to make the VTD-XML freely available to all users, including Ameriprise and its contractors. Ameriprise's counterclaim alleging violations of the GPL was sent back to the state court because it was beyond the scope of the Copyright Act, thus leaving unresolved the question of whether software companies can restrict their contractors from further redistributing their GPLv2 license. (*Versata Software, Inc. v. Ameriprise Financial, Inc.,* No. A-14-CA-12-SS (W.D. Tex. Mar. 10, 2014).

Finally, it's important to note efforts by companies to avoid litigation over failure to comply with open source licenses. In 2017, Facebook, Google, IBM, and Red Hat agreed to extend the GPLv3s 60-day cure period to address compliance errors to the other GPL licenses, including GPLv2 and LGPLv 2.1 and v2. The 60-day cure period allows licensees to comply with the GPL provisions prior to litigation or termination of the agreement.

5.2 Is there any foreign case law addressing OSS licensing?

Two German courts have enforced the GPL license terms against several foreign vendors that have not made modified source code available after incorporating OSS in their products offered for commercial sale. (Welte v. Sitecom, Final Judgment of the District Court of Munich I, issued 19 May 2004 – Docket No. 21 O 6123/04; Welte v. D-Link Germany GmbH, District Court (Landgericht) of Frankfurt Am Main, Docket No. 2-6 0 224/06; Welte v. Skype Technologies S.A., District Court (Landgericht) of Munich I, Docket No. 7 O 5245/07.) A variant of the antitrust argument used unsuccessfully in the U.S. in Wallace v. IBM was also put forth initially on appeal of Skype in Germany, but the appeal was subsequently withdrawn, so the German judicial view of this argument remains unknown.

An assignation was filed before a French court (le Tribunal de Grande Instance de Paris) in late November 2008, against the French telecom company, Iliad, on behalf of the Free Software Foundation, Mr. Harald Welte, and others. The complaint alleges that Iliad incorporated GPL-licensed software into its Freebox products, which were then distributed to the public without making modified source code available. (No citation is provided for this case because only an unauthenticated copy of the assignation document is available via Internet search engine sources.)

In 2015, Linux kernel developer Christoph Hellwig, supported by the Software Freedom Conservancy, filed a copyright infringement suit against software giant VMWare in a German district court alleging that the company's combining his contributions to the Linux kernel with its own proprietary code vmkernel and distributing the entire product as a commercial product violated the GPLv2 provisions requiring users to distribute derivative works. Because the claims required the court to interpret the scope of the GPL and specifically the extent of its copyleft requirement or the "derivative work" issue when open source and proprietary code is combined, the case attracted worldwide attention. The district court dismissed the case on the basis of evidentiary flaws in that the Complainant failed to clearly identify the specific lines of code he authored. The case was also dismissed on appeal for the same reason. Helwig decided not to appeal the case further because VMware announced that it would discontinue using the code, thus finally complying with the GPL. As to the future impact of the decision on rights holders and their decision to seek judicial review, Helwig notes that "at least in Germany, GPL violations that cannot be resolved out of court will probably require greater involvement of rights holders. If -- as is typical with the Linux kernel -- numerous rights holders exist, individual developers will find it difficult to go to court alone.[14]"

In a recent software infringement case involving two Chinese companies, Plaintiff Pomelo (Beijing) Technology CO., LTD (DCloud) & Defendant Pomelo (Beijing) Mobile Technology C)., LTD. (APICloud), the court affirmed the enforceability of the GNU General Public License version 3 (GPL-3.0), especially the copyleft mechanism.[15] The case was filed in 2015 and decided in April of 2018. The plaintiff alleged that the defendant copied and adapted three independent plug-ins of the plaintiff HBuilder software development kit into the defendant's released APICloud toolset. Relying on the copyleft mechanism as their primary defense, the defendants argued that the HBuilder project as a whole should be made publicly available under the GPL-3.0 license. In ruling against the defendants, the court held that the GPL license text does not apply to the plug-ins at dispute, thus they are not derivative works or modifications referred to in the license and are not required to be made publicly available.

5.3 Other Legal Issues Raised under OSS Licensing

Open Source Security, Inc. v. Perens, No. 17-cv-04002-LB (N.D. Cal. June 9, 2018).
Open Source Security, Inc. (OSS) was a provider of security software code ("Patches") under the trade name of Grsecurity® for the Linux® Operating System. The Patches were released under the GNU General Public License, version 2 ("GPLv2") and contingent upon a subscription agreement. The agreement stated a policy of terminating access to future updates if a user redistributes Patches sets or changelogs "outside of the explicit obligations under the GPL to User's customers," and in 2015 began limiting access to its software code to paying customers only. On June 28, 2017 Bruce Perens published a blog post contending that the Grsecurity agreement violated the GPL thereby exposing OSS customers to risk for breach of contract and copyright infringement[16]. OSS sued Perens in the Northern District of California, claiming that Perens' blog has resulted in substantial harm to its reputation, goodwill, and future business prospects. In December 2017, the court dismissed the complaint saying that statements of opinion by a person who is not an attorney, were not defamatory. The courts have not yet addressed this legal issue, so Perens' opinions are not actionable libel.
A month later in January 2018, Perens sued OSS to recover attorneys' fees and costs pursuant to California's anti-SLAPP statute, California Code of Civil Procedure Section and was awarded $259,900.

[14] https://www.zdnet.com/article/linux-developer-abandons-vmware-lawsuit/

[15] Filed and ruled on in China's Intellectual Property Right Court in Beijing.

[16] Section 6 of the General Public License forbids users who redistribute the Linux kernel from restricting its use: "Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein." https://www.gnu.org/licenses/old-licenses/gpl-2.0.en.html

6 Considerations on Use of OSS: Tips and Best Practices

6.1 When should I discuss OSS with my clients?

OSS should be discussed in detail whenever your client is considering: (1) the incorporation of OSS into software developed by or for the agency; or (2) the original development by or for the agency of software intended for open source release. It is a best practice to always include legal counsel early in considerations.

OSS should be discussed more generally whenever your client is considering: (1) acquiring or using (or modifying) a new computer program or computer-based technical data; (2) modifying current computer programs or technical data; or (3) acquiring or using or designing computer networks or hardware that contains embedded software that may ultimately be linked to non-OSS software or technical data.

If a proposed release of software developed by or for the agency includes the release of OSS, care must be taken to ensure that the pertinent license for such OSS is acceptable. Counsel should review the OSS license and assess any special risks that may be involved, and confirm that the agency has obtained clear rights from any third party rights owners (such as through an assignment or license) to make the Open Source Release. For example, at least one widely used OSS license requires that all software distributed with that OSS be distributed under the same license terms.

If the specific OSS provisions and responsibilities will not allow the client to accomplish its mission, the client, with the assistance of counsel, can attempt to take action to negotiate new terms directly with the OSS owner. See the last paragraph under 4.8 for limitations.

6.2 What do clients need to understand before making a programmatic or acquisition decision?

At a minimum:

- Clients must understand exactly what types of software (OSS and non-open source commercial/non-commercial software) are proposed for delivery, use or work performance.
- Clients must know how each type of OSS will be used during development, as well as in any delivered software.
  - The government must be prepared to accept delivery of OSS under the terms of the OSS license. On the other hand, software may be proposed for use in performance but not delivery under a contract. In this case the client must take care that such use does not create any additional government obligations
- Clients must know where each type of OSS will be used during development, in delivered code, and for use on which computer networks or systems.
  - This can present issues for government applications in some circumstances, where distribution of the source code might be required by the software

license, but may be forbidden by other circumstances, such as export control, classification, proprietary data rights, or other national security interests. Legal and security staff can provide guidance.

- Clients must understand the terms of each OSS license. Some commercial software licenses, including OSS licenses, include terms unacceptable to the government, such as, certain kinds of indemnification, choice of law, choice of forum, reimbursement of attorneys' fees, etc.
- Clients must know whether the OSS has been or will be modified during development or after delivery to the government.

Noncompliance with OSS license terms could result in litigation or loss of use of the software.

6.3 How can my agency identify the OSS it may already be using if the software does not already carry identifying markings that reflect it is OSS?

Other than asking the software developer for the code provenance, or by doing an extensive manual examination of the source code (which would likely not yield conclusive results anyway), experts differ on whether it is presently feasible to determine reliably whether OSS code may be present in computer software already delivered to, or developed by the government, unless the developer has previously labeled all or part of the existing code as an identifiable version of OSS. Several commercial companies offer examination services for this purpose.

6.4 Should the government care whether its contractors identify OSS they may be embedding in or linked to software delivered to the government under procurement contracts, cooperative agreements, or other instruments?

Yes. Some OSS licensing provisions can directly affect whether and when the government may be obliged to provide source code to the public. Agency procurement officials should consider including notices in Requests for Proposal and contracts for software development regarding:

(1) whether OSS should or may be used in software developed for or delivered to the government;
(2) requirements that the contractor identify any OSS that may be incorporated into software developed for or delivered to the government; and
(3) requirements that the contractor provide copies of all OSS licenses.

6.5 Should the government care whether its contractors use OSS products to develop software for the government, but do not embed any OSS in any of the delivered code?

Yes. Some OSS licensing provisions can directly affect whether and when the government may be obliged to provide source code to the public, even when the OSS is not embedded in the delivered code, but is only used in its development. If the contractor uses OSS in the performance of a government contract, it must ensure that its use does not create any government distribution obligations with respect to the computer software deliverables; or grant to any third party any rights to or immunities under government intellectual property or government data rights to the computer software deliverables.

6.6 Can the government require contractors to tell the government about OSS they are using in code that is licensed or delivered to the government under government contracts, or that is used to develop such code?

Yes. But even if government contractors do not volunteer this information, government agencies should always request all contractors to identify fully to the government their intended uses and planned modifications of OSS that are expected during their performance of a government contract, whether or not the OSS used or generated will be delivered to the government. As with all other commercial software code, the government should ask the contractor to clearly identify in writing all of the following items:

(1) each type of OSS used/modified and its title and version number;
(2) each concomitant OSS license and, if applicable, license version number;
(3) identity of the asserting party (contractor/sub/awardee);
(4) whether any of the OSS has been or will be modified, and, if so, by whom; and
(5) whether such modification occurred or will occur by incorporating it into any third party software (if so, identify).

While this information should be provided for all commercial software (including OSS) prior to execution of any contract or award instrument, the parties should also agree that full written identification of all commercial code used, including OSS, must be provided by the contractor and approved by the government before incorporating it into any deliverable, using it to develop a deliverable, or using it to modify or link to preexisting code used in any government computer program or system.

6.7 Is government-created software considered an agency record covered by the Freedom of Information Act (FOIA)?

Generally, no, because software used by federal agencies is typically obtained through a license that limits, by its terms, the agency "control" of the software that would be necessary for it to be an "agency record" subject to the FOIA. In rare cases, however, courts have found software used and possessed by federal agencies to be agency records when the software was "uniquely suited to its underlying database" such that "the software's design and ability to manipulate the data reflect the [agency's study] …." *Cleary, Gottlieb, Steen & Hamilton v. HHS*, 844 F. Supp. 770, 781-82 (D.D.C. 1993). In short, the more the software itself and the agency's unique use of the software reflect agency operations, the more likely the software will be considered an agency record subject to the FOIA.

Specific examples of computer software programs that may be agency records subject to the FOIA include: (1) software featuring an embedded database subject to the FOIA that cannot be extracted from the software; (2) software that directly reveals information about agency policy, functions, decision making, or procedures; and (3) software that cannot be separated from an accompanying database subject to the FOIA without rendering the database unintelligible or unusable. In these instances, both the data and the software must be reviewed for potential release or withholding under the FOIA.

Other cases addressing this issue include, *Gilmore v. Department of Energy*, 4 F. Supp 2d 912 (N.D. CA 1998 (N.D. CA 1998), and *DeLorme Pub. Co. v. NOAA*, 907 F. Supp. 10 (D. Me 1995).

6.8 Does the licensing of OSS raise export control issues?

Yes, if OSS that requires an export license under the Export Reform Control Act 50 U.S.C. Chapter 58 or is on the U.S. Munitions List (22 CFR § 121.1) is released to a foreign person, academic institution, company, government, or nongovernmental organization, violations of the Export Reform Control Act, the Arms Export Control Act (AECA) (22 U.S.C. §§ 2778-2780) and the International Traffic in Arms Regulations (ITAR) (22 C.F.R. Parts 120-130) may occur. Generally, the Export Reform Control Act grants the President authority to control: (1) the export, reexport, and transfer of items (commodities, software, or technology), whether by U.S. persons (including corporations) or by foreign persons, wherever located to protect national security; and (2) the activities of U.S. persons, wherever located, relating to specific nuclear explosive devices, missiles, chemical or biological weapons, whole plants for chemical weapons precursors, foreign maritime nuclear projects, and foreign intelligence services. The AECA and ITAR regulate defense articles and services and related technical data identified on the U.S. Munitions List. Additional information on licensing requirements, policies and procedures related to export controls may be found on the U.S. Department of Commerce's and Department of State's web sites at http://www.bis.doc.gov and http://www.pmddtc.state.gov. Recently, on Jan. 6, 2020, the U.S. Government amended its dual-use export controls to cover certain software used in Artificial Intelligence (AI) applications. Specifically, the Bureau of Industry and Security (BIS) amended the Export Administration Regulations (EAR) to impose license requirements on the export and reexport of U.S.-origin software specially designed to automate the analysis of geospatial imagery to all destinations, except Canada. Other U.S. Agencies also regulate the export of certain goods and services, for example the Department of Energy, the Nuclear Regulatory Commission; and the Treasury Department Office of Foreign Assets Control administers and enforces trade embargoes and sanctions. Exercise caution if the OSS contains encryption technology[17]. Consult agency counsel with expertise in export control matters for guidance on licensing OSS that may raise export control issues.

7 Legislation and Other Resources

The bibliography lists some recent publications, articles, brochures, web sites, related to computer software copyright that provide information and a variety of perspectives on this issue. This list is not intended to be exhaustive nor does the U.S. Copyright Office necessarily endorse the works listed. Cited web site addresses were all correct and active as of March 2020.

[17] See Bureau of Industry Security website "Encryption and Export Administration Regulations"
https://www.bis.doc.gov/index.php/policy-guidance/encryption

7.1 Web Sites

These web sites contain references, links, and additional informational resources and opinions on copyright as it relates to computer software. Many of these sites have links to other informational materials with related OSS themes.

Code.gov Primary government platform for sharing open source code.
https://code.gov/#/

Code.mil is an experiment in open source at the Department of Defense. The goal is to foster open collaboration with the developer community around the world on DoD open source projects.
https://code.mil/

DARPA Open Catalog - curated list of DARPA-sponsored software and peer-reviewed publications.
https://opencatalog.darpa.mil/

Department of Transportation

Department of Transportation Intelligent Transportation Systems Joint Program Office Code https://its.dot.gov/code/

Open Source Application Development Portal (OSADP) This is a web-based portal that features source code, software, applications, and resources (e.g., documentation, licenses, data) to support the use of or further development of ITS(Intelligent Transportation Systems)-related applications.
https://www.itsforge.net/

DoD Open Source Software (OSS) FAQ - This page is an educational resource for government employees and government contractors to understand the policies and legal issues relating to the use of open source software (OSS) in the Department of Defense (DoD).
https://dodcio.defense.gov/Open-Source-Software-FAQ/

DOE CODE - Collaboration, archiving, and discovery of scientific and business software.
https://www.osti.gov/doecode/

Forge.mil Program - Family of services provided to support the DoD's technology development community. The system enables the collaborative development and use of open source and DoD community source software.
http://www.forge.mil/

GSA Federal GitHub Dashboard - Code discoverability and usability for developers looking to reuse what the federal government has already developed.
http://gsa.github.io/github-federal-stats/index.html

Health.mil The Military Health System's sampling of some Open Source Software available to the public.
https://health.mil/

NASA - Portal for NASA open source software.
https://code.nasa.gov/

National Institutes of Health – software repositories for bioinformatics, genomics and life science.

https://www.nihlibrary.nih.gov/services/bioinformatics-support/computational-molecular-biology-curated-list/software-repositories

National Library of Medicine - NIH-supported data repositories that make data accessible for reuse. Most accept submissions of appropriate data from NIH-funded investigators (and others), but some restrict data submission to only those researchers involved in a specific research network. Also included are resources that aggregate information about biomedical data and information sharing systems.

https://www.nlm.nih.gov/NIHbmic/nih_data_sharing_repositories.html

National Security Agency – open source software developed by NSA and available to the public.

https://code.nsa.gov/

Open Source Initiative (OSI)
www.opensource.org

Source Forge
sourceforge.net

U.S. Copyright Office
www.copyright.gov

United States Patent and Trademark Office
www.uspto.gov

18F
https://18f.gsa.gov/open-source-policy/
Facts about publishing open source code in government

7.2 Other Sources (Publications, Reports etc.)

The American Bar Association
https://www.americanbar.org/aba.html
Search for "open source software"

Free Software Foundation (FSF)
https://www.fsf.org/

General Public Licenses for Open Source Software – Video tutorials from the DoD Enterprise Software Initiative (DoD ESI)
http://www.esi.mil/videos.aspx#

Mitre/DISA, Use of Free and Open-Source Software (FOSS) in the U.S. Department of Defense, Version 1.2.04, January 2, 2003
http://dodcio.defense.gov/Portals/0/Documents/FOSS/dodfoss_pdf.pdf

This document does not constitute legal advice and should not be construed or used as such.