

13

Open Standards

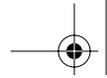
Defining Open Standards

The first presentation I ever attended about open source was actually supposed to be about *open standards*. A panel of representatives from some major software companies was trying to define *open standards* for the audience. They couldn't agree on a definition, and they kept confusing *open standards* with *open source*.

By then I had already started working with Open Source Initiative and I was smugly confident about the definition of *open source*. We had a published Open Source Definition to rely on (see Chapter 1). I understood the relationship between *open source* and *software freedom*. But I hadn't the slightest clue what the panelists really meant by *open standards*. Was it somehow also related to *software freedom*?

I believed then, even if this panel wasn't explaining it well, that at least the venerable standards bodies around the world must have found a way for standards to be "freely implemented" worldwide. It turns out that we were all a few years too early. Not until 2002 was an effective definition of *open standards* published by the World Wide Web Consortium (W3C) that was truly compatible with open source. I'll reprint that definition in full later in this chapter.





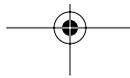
Standards are developed by industry consortia that, within the guidelines of the antitrust laws, cooperate to publish specifications for how products should interoperate. A common design or implementation is often essential to help prevent fragmented development of products that don't work with each other. Each participating company is expected to satisfy the agreed specifications in its products—and each company is encouraged to seek its own way to improve upon the specifications and to distinguish its own products from those of its competitors. As Scott Peterson from Hewlett Packard once described it to me, “Companies cooperate on standards and compete on implementations.”

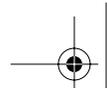
We couldn't live without industry standards. Standards allow telephones from one manufacturer to work on the communications switches of other manufacturers. All browsers (at least in theory) can display web pages identically if they meet industry standards. Electronic mail systems from different software vendors can exchange email. Without standards, this would truly be a Tower of Babel world.

Open Specifications

Suppose someone writes a book that teaches how to calculate income taxes, a specification for a yearly process that you hate to do manually. You read the book at your local library. You then implement the specification in computer software, creating your own original copyrightable work. You do not copy the book in your software, except perhaps in a few places where it says things like “subtract your deductions from your gross annual income” and you translate that into source code within your software. Are you a copyright infringer?

Colloquially, we often say things like “You *copied* the specification.” But this has little to do with the definition of *copy*





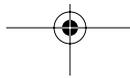
that I explained in Chapter 2. What we often mean to say is, “You read and understood what the specification told you about income tax rules and procedures and then, starting from scratch but relying on what you learned, you wrote your software.” If you copied anything, it was the book’s underlying ideas—what I have already described as “uncopyrightable subject matter.”

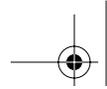
In other words, the copying that you did when you implemented the standard is not necessarily copyright infringement. You do not appropriate the copyrightable intellectual property of the specification’s author by implementing your software without directly copying the specification.

The specification document itself, of course, the book that was published by the standards organization, is copyrighted. That specification meets the definition of both “original work of authorship” and “copy” from the U.S. Copyright Act. (17 U.S.C. § 101.) The specification cannot be copied without the copyright owner’s permission. (17 U.S.C. § 106.)

Simply because it describes an open standard does not mean that you can make copies or distribute that specification. You have to look separately to the specification license to determine whether you may do so. (This is no longer true in some jurisdictions for specifications that are incorporated by reference into laws and are enforceable under the law.) In general, the owner of the copyright to the specification—perhaps the standards organization itself, or one or more of its members—can license the specification in any way.

A specification license that prohibits people from reading the specification without paying a license fee to the licensor, or that restricts in any way the use of the information it contains, is not an open specification license. It is incompatible with Open Source Principle # 1. Such standards are not open standards.





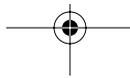
A published specification describing an open standard, just like open source software, need not be distributed at zero price. (See Open Source Principle # 2.) Standards organizations can and some do sell copies of their specifications. Because the goal of most standards organizations is to maximize the implementation of their specifications, most often do not overcharge—or charge at all—for their documents.

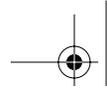
Some standards organizations recover their costs by selling copies of their specifications and, when the cost is reasonable, most people will pay for official printed copies. Whichever the pricing model and whatever the price of a single copy of the specification of an open standard, any number of people can read that copy. So also may any number of people write software that implements that specification without any further payments to the copyright owner of the specification.

Enforcing the Standard by Copyright Restrictions

Some standards become laws that everyone must obey. For example, in the United States there are uniform codes regulating the building, electrical, and plumbing trades. Contractors may not build things that violate those industry standards.

Many state and local laws mandate industry standards without actually stating the standard; they incorporate the standard by reference to some specification published by a standards organization. These standards have the force of law and must be obeyed. Courts in the United States have only recently addressed the issue of standards organizations being able to charge fees for the public's right to copy industry standards that are enforceable under the law. In some jurisdictions, royalties for the right to make copies of laws are no longer allowed.





Some companies and other nongovernmental organizations also want to control industry standards. Since those industry standards are not adopted by legislatures as laws, they cannot be enforced like building, electrical, and plumbing codes. Private owners of the intellectual property in standards can enforce their standards privately, under contract law and through the application of copyright, patent, and trademark law, by controlling license rights to the specifications of the standards.

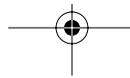
As described below, some of those copyright, patent, and trademark licenses are compatible with open source and open standards.

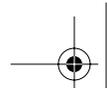
Licensing the Test Suite: The Open Group License

The Open Group is a standards organization that promotes, among other things, standards relating to UNIX. It also owns the UNIX certification mark that is registered around the world, and it manages a program to certify UNIX implementations by other companies. Versions of UNIX that meet the Open Group's specifications may carry the UNIX certification mark.

Certification requires testing. Under trademark law in most countries, the certifying organization must ensure that its certification marks are used only on tested and approved products. Otherwise the certification mark may be lost. A certifying organization (e.g., The Open Group) is responsible for verifying the quality of the certified goods.

The Open Group does this through published test suites, programs that are used to test versions of UNIX. If the test suites run successfully on the to-be-certified UNIX implementation, that UNIX version is certified.





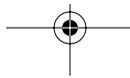
The Open Group Test Suite License is for the test suite software itself, the Package. (See *www.opensource.org* for a copy of this license.) That Package is open source. The license does not require that the UNIX implementations that are tested against that Package themselves be open source.

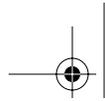
The Open Group Test Suite License seeks to control the copyrightable elements of the test suite software sufficiently to protect the Open Group's certification marks. The preamble to the license calls it "artistic control" but this license actually has a much more practical objective. The Open Group is primarily concerned with the importance of testing to ensure conformance to the standards:

Since these are benchmark measures of conformance, we feel the integrity of test tools is of importance. In order to preserve the integrity of the existing conformance modes of this test package and to permit recipients of modified versions of this package to run the original test modes, this license requires that the original test modes be preserved. (Open Group Test Suite License Preamble.)

This license conforms to the Open Source Principles. Licensees may copy, modify, and distribute copies of the Package. These are the important conditions:

- You must duplicate all of the original copyright notices and associated disclaimers from the Standard Version of this Package. (Open Group Test Suite License section 1.)
- You must insert a prominent notice in each changed file stating how and when you changed that file. (Open Group Test Suite License section 3.)





- You must rename any nonstandard executables and test cases and provide a separate manual page that clearly documents how it differs from the Standard Version. (Open Group Test Suite License section 3.)
- When you distribute your version, you must accompany your modifications with their corresponding Standard Version executables and test cases. (Open Group Test Suite License section 4.)

Through this open source license on its test suite Package, The Open Group is able to control the standards for its own certification mark while granting to everyone the software freedom to create derivative works of the Package. Those derivative works are not required to comply with the standard, but if they do not they cannot be called the *Standard Version*:

“Standard Version” refers to such a Package if it has not been modified, or has been modified in accordance with the wishes of the Copyright Holder. (Open Group Test Suite License Definitions.)

Only those UNIX implementations that successfully passed the Standard Version tests will be certified by the Open Group to call themselves UNIX.

Discouraging Forks: Sun’s SISSL

Sun Microsystems wanted a more robust way to prevent the standard from being forked. *Forking* is a colloquial term used in the open source community to describe what happens when a cooperative project splits into two or more uncooperative separate projects. The result is either an opportunity or a prob-





lem, depending partly on whether you're the project being forked from or to, and partly on the ultimate success of the forked project's software in the marketplace. One of the risks of permitting derivative works of industry standard specifications and test suites is that competitors may move away from the standard. As I just described, The Open Group Test Suite License avoided that by requiring notice and documentation of such changes, and it prohibited calling derivative works the *Standard Version*. But that's only partially effective. Companies can diverge from the standard, or add new requirements, without having to return those contributions to the open standard.

An open source license cannot prohibit forks. (Refer to Open Source Principle # 3, which mandates the freedom to create derivative works.) But the license can set conditions, including a reciprocity condition, on such derivative works.

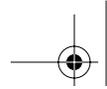
The Sun Industry Standards Source License (SISSL) is patterned largely on the MPL, with its emphasis on files rather than the broader concept of derivative works. (The full text of the SISSL is available at www.opensource.org.) You will recognize much of the MPL's structure, with this interesting addition to the reciprocity condition.

The Modifications which You create must comply with all requirements set out by the Standards body in effect one hundred twenty (120) days before You ship the Contributor Version. In the event that the Modifications do not meet such requirements, You agree to publish either

(i) any deviation from the Standards protocol resulting from implementation of Your Modifications and a reference implementation of Your Modifications or

(ii) Your Modifications in Source Code form, and to make any such deviation and reference implementation or Modifications available to all third parties under the same terms





*as this license on a royalty free basis within thirty (30) days of Your first customer shipment of Your Modifications.
(SISSL license section 3.1.)*

Like all reciprocity provisions in open source licenses, the SISSL requires no more of the licensee than the licensor already gave. It permits forks of the standard, but any Modifications that break compatibility with the standard will be available on a reciprocal basis for all to adopt. It also imposes timing constraints on the creation of derivative works that allow the standards organization—in this case Sun Microsystems—an opportunity to react to attempted forks.

Sun uses the SISSL license for the file format and application programming interface specifications of its version of Open Office software, and the GPL for the Open Office software itself.

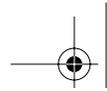
Patents on Open Standards

What happens when someone owns patents that are necessary to implement the specification for an open standard? You will recall that the owner of a patent can prevent you from making, using, or selling his or her patented invention regardless of how you learned to do it, even if you invented it yourself subsequently.

If someone owns a patent claim necessary to practice an open standard, you will need a license from the patent owner to practice that standard in your own software. Your freedom to practice the standard in your software is subject to the license terms from the patent owner.

Standards organizations recognize this. That is why they have focused in recent years on designing patent policies that are compatible with open source. The key to open standards is a patent policy that encourages the widespread adoption of the





standard in all kinds of software—including open source software.

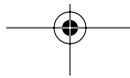
Patent claims necessary to practice an industry standard can suddenly appear. The story is often written about the eccentric scientist who, while puttering in his garage, secretly invents and perhaps tries to delay the publication of an essential patent to valuable technology. There is nothing that a standards organization, or anyone else, can do to prevent such surprise patents that are published by the Patent Office after a standard is promulgated.

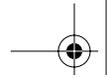
But far more typically, important patents are owned by the same companies that participate in the standards organizations. Who, after all, is more likely to want to file patents in a particular industry technology than the companies that have special expertise in that field? Those companies have the talent and resources to create a wealth of patents surrounding the field of the standards.

Standards organizations need ways to protect their members from each others' private patents. The latest technique, the development of agreed patent policies that limit the options of their members to enforce private patents, is one important solution to the patent problem for industry standards. The patent policy of the W3C is the leader in this new area of open standards; the W3C Patent License is described in the last section of this chapter.

Reasonable and Nondiscriminatory

Most standards organizations demand that their members agree to license any of their patent claims necessary to practice their standards on “reasonable and nondiscriminatory terms.” Here is a typical license grant from one company, Cisco, to one standards organization, the Internet Engineering Task Force:





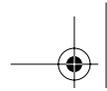
Cisco has a pending patent application relating to the subject matter of draft-ietf-mobileip-nat-traversal-06.txt, "Mobile IP NAT/NAPT Traversal using UDP Tunneling". If a standard relating to this subject matter is adopted by IETF and any claims of any issued Cisco patents are necessary for practicing this standard, any party will be able to obtain a license from Cisco to use any such patent claims under reasonable, nondiscriminatory terms, with reciprocity, to implement and fully comply with the standard. (From www.ietf.org.)

The key words in this letter are *reasonable* and *nondiscriminatory*. You will see these words in most patent grants to most standards organizations worldwide. This is just one example; Cisco and the IETF are not unique. I'm not picking on them by reprinting this letter.

The word *reasonable* is impossible to define precisely. It always depends on the facts of the specific case. So, for example, there is no single reasonable price for a car or a house, no agreement on what constitutes reasonable warranty terms, and perhaps for some companies there is no reasonable way at all to accept a reciprocity provision. What is the reasonable jurisdiction and venue for litigation against an open source programmer who lives in Africa or Europe?

The word *nondiscriminatory* is also ambiguous. Does it mean that both rich and poor will not be discriminated against? (It is difficult to set any price other than very near zero that doesn't discriminate against at least some of the poor.) Or does the promise not to discriminate merely extend to the forms of discrimination already outlawed by law, such as age, race, and sex? As some have complained about the GPL and other reciprocal open source licenses, aren't all reciprocity provisions discriminatory against those who won't or can't accept a reciprocity obligation?





In practice, the reasonable and nondiscriminatory promises simply mean that everyone will pay the same price, and be subject to the same terms and conditions, for the same patent license rights—even if those terms and conditions are onerous and incompatible with free software. That is not open source, any more than saying that Microsoft Windows is open source because everyone pays the same price and agrees to the same End User License Agreement. As I have noted throughout this book, the devil is in the detailed license terms and conditions that must be agreed to.

Another ambiguous phrase in the Cisco letter is *with reciprocity*. The scope of the reciprocal license expected from implementers or users of the standard is unknown until the precise license terms are revealed by Cisco. Is reciprocity in this case benign?

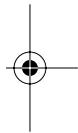
An open source licensor can take little comfort when a company issues vague promises of reasonable and nondiscriminatory licenses for its patents. We need to be certain that the patent licenses are actually compatible with open source.

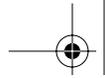
Royalty Free

Software freedom doesn't require zero price for a copy of the specification describing how to write software. But it does require zero royalties for a license to those patent claims necessary to make, use, and sell open source implementations of that software. A price other than zero for the right to make copies conflicts directly with Open Source Principle # 2.

Therefore, the only reasonable royalty for a patent license for an open standard that can be implemented in open source is zero. The term of art for such a license is *royalty free*.

Very few of the reasonable and nondiscriminatory patent licenses for industry standards actually charge a royalty. As a





practical matter, the word *reasonable* mostly means *zero*. But not always, and when a license requires payment of a royalty, it poses a problem for open source software developers who can't recover that royalty through license fees.

It would be a mistake, though, to just focus on price. As I have described throughout this book, there are many other characteristics of open source software that matter much more, such as the right to create derivative works. An open standard patent license that is compatible with open source must include more than a promise of a zero royalty.

The term *royalty free* is now potentially as confusing as the term *free* was for software. Perhaps it would be better if we called standards that satisfy the W3C Royalty-Free Patent License requirements *open standards*?

The W3C Patent License

The World Wide Web Consortium was the first software industry standards organization to confront directly the problem of patent licenses for open source software. In May 2003, following several years of internal debate among W3C members (including representatives from all the major software companies and open source organizations), W3C published its patent policy. The effort was characterized by W3C director Tim Burners-Lee as “the most thorough ... to date in defining a basic patent policy for standard-setting.” (See www.w3.org.)

One of their major goals was to make W3C standards (what they call *Recommendations*) fully compatible with open source software.

As a condition for participating on a specific W3C standard-setting working group, W3C member companies and their representatives undertake to disclose and/or license their patents relating to that working group to everyone under an open





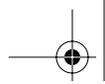
source compatible patent license. A member company can refuse to license its patents for a W3C standard. But if it fails to disclose the existence of those patents, or if it decides to issue licenses, it must license its patents under a license compatible with the W3C Patent Policy.

These are the requirements for such patent licenses:

With respect to a Recommendation developed under this policy, a W3C Royalty-Free license shall mean a non-assignable, non-sublicensable license to make, have made, use, sell, have sold, offer to sell, import, and distribute and dispose of implementations of the Recommendation that:

- 1. shall be available to all, worldwide, whether or not they are W3C Members;*
- 2. shall extend to all Essential Claims owned or controlled by the licensor;*
- 3. may be limited to implementations of the Recommendation, and to what is required by the Recommendation;*
- 4. may be conditioned on a grant of a reciprocal RF license (as defined in this policy) to all Essential Claims owned or controlled by the licensee. A reciprocal license may be required to be available to all, and a reciprocal license may itself be conditioned on a further reciprocal license from all.*
- 5. may not be conditioned on payment of royalties, fees or other consideration;*
- 6. may be suspended with respect to any licensee when licensor is sued by licensee for infringement of claims essential to implement any W3C Recommendation;*
- 7. may not impose any further conditions or restrictions on the use of any technology, intellectual property rights, or other restrictions on behavior of the licensee, but may include rea-*





sonable, customary terms relating to operation or maintenance of the license relationship such as the following: choice of law and dispute resolution;

*8. shall not be considered accepted by an implementer who manifests an intent **not** to accept the terms of the W3C Royalty-Free license as offered by the licensor.*

License term:

9. The RF license conforming to the requirements in this policy shall be made available by the licensor as long as the Recommendation is in effect. The term of such license shall be for the life of the patents in question, subject to the limitations of 5(10).

10. If the Recommendation is rescinded by W3C, then no new licenses need be granted but any licenses granted before the Recommendation was rescinded shall remain in effect.

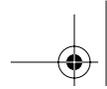
(See www.w3.org.)

Of particular importance, of course, are items 1, 5, and 7, which allow everyone to make, use, or sell standard open source software, and which prevent the imposition of patent license conditions that would restrict its creation or distribution. Such licenses are compatible with the Open Source Principles from Chapter 1.

The W3C Royalty-Free license is a model for open standards patent licenses that are compatible with open source. Other standards organizations are beginning to consider similar licensing models.

Not every requirement of the W3C Royalty-Free license policy is friendly to open source, however. For example, because such licenses are “non-assignable” and “non-sublicenseable,” each licensee theoretically must obtain a license directly from the patent owner. In practice hardly anybody





does, and because of the W3C member commitments to each other, nobody needs to fear that a royalty-free patent license wouldn't be available to anyone who actually wanted one.

Item 3 allows the imposition of a field of use restriction in a patent license. Everyone should recognize that in some situations this field of use restriction may limit the creation of certain types of derivative works. This is not a unique problem for the W3C patent license; remember that open source licenses such as the MPL and CPL also contain subtle but important field of use restrictions.

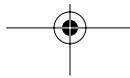
Item 6 allows the patent being licensed to be used for defensive purposes. Anyone who sues the patent owner for patent infringement risks having patent licenses to “this and other W3C specifications” suspended (or terminated). Similar provisions in many open source licenses have already been discussed in this book. Open source licensors are allowed to use their intellectual property to defend against infringement lawsuits by others.

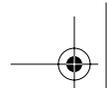
Justifying Open Standards and Open Source

Item 5 of the W3C Royalty-Free license, the requirement that a patent license “may not be conditioned on payment of royalties, fees or other consideration,” is the most significant factor for most companies. They face the prospect of licensing some of their patented intellectual property at zero price if they contribute to the development of an industry standard.

How could contributing patents at zero price for open standards ever be justified to company shareholders?

Somehow it must be justified over and over again, because very few companies actually charge royalties for their patent licenses relating to industry standards. Zero price is typical even though it is not yet generally the rule. Companies have





long recognized that charging royalties for some things will impede the beneficial cooperation for which they joined industry consortia in the first place. It is better to forego small royalty profits for a small number of patents in exchange for the prospect of long term financial gain in a vibrant, competitive marketplace.

This is the same economic tradeoff that confronts a copyright licensor who is considering licensing software under an open source license. The licensor's customers will be able to make unlimited free copies of this copyrighted intellectual property. How can a licensor make money that way?

You will find many examples of profitable open source business models among the major software companies and open source projects worldwide. We now see huge collections of open source software being created and contributed to around the world under the licenses described in this book. The price of software copyright and patent licenses isn't always the most important characteristic or advantage of open source software.

Open source and open standards are an enormous reality even if this book doesn't fully explain why people and companies do it. I could only describe licensing in this book. I could not also help you to justify the underlying open source business models. That is for someone else's book.

The simple fact is that many companies and individuals now contribute to a growing commons of intellectual property. They have discovered that more value is derived by distributing this intellectual property freely to others and sharing in the growing public commons of free software.

