McCoy Smith
Intel Corporation, Law & Policy Group
Open Source
Open Source

• A Philosophy

• A Standard for Evaluating Software Licenses

• A Set of Community Expectations
Open Source

• A Philosophy

What is free software?

The Free Software Definition

The free software definition presents the criteria for whether a particular software program qualifies as free software. From time to time we revise this definition, to clarify it or to resolve questions about subtle issues. See the History section below for a list of changes that affect the definition of free software.

"Free software" means software that respects users' freedom and community. Roughly, it means that the users have the freedom to run, copy, distribute, study, change and improve the software. Thus, "free software" is a matter of liberty, not price. To understand the concept, you should think of "free" as in "free speech," not as in "free beer." We sometimes call it "libre software" to show we do not mean it is gratis.

We campaign for these freedoms because everyone deserves them. With these freedoms, the users (both individually and collectively) control the program and what it does for them. When users don't control the program, we call it a "nonfree" or "proprietary" program. The nonfree program controls the users, and the developer controls the program, this makes the program an instrument of unjust power.

A program is free software if the program's users have the four essential freedoms:

• The freedom to run the program as you wish, for any purpose (freedom 0).
• The freedom to study how the program works, and change it so it does your computing as you wish (freedom 1). Access to the source code is a precondition for this.
• The freedom to redistribute copies so you can help your neighbor (freedom 2).
• The freedom to distribute copies of your modified versions to others (freedom 3). By doing this you can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this.

A program is free software if it gives users adequately all of these freedoms. Otherwise, it is nonfree. While we can distinguish various nonfree distribution schemes in terms of how far they fall short of being free, we consider them all equally unethical.
Open Source

• A Philosophy

The Free Software Definition

The free software definition presents the criteria for whether a particular software program qualifies as free software. From time to time we review this definition, to clarify it or to resolve questions about subtle issues. See the History section below for a list of changes that affect the definition of free software.

"Free software" means software that respects users' freedom and community. Roughly, it means that the users have the freedom to run, copy, distribute, study, change and improve the software. Thus, "free software" is a matter of liberty, not price. To understand the concept, you should think of "free" as in "free speech, not as in "free beer". We sometimes call it "libre software" to show we do not mean it is gratis.

We campaign for these freedoms because everyone deserves them. With these freedoms, the users (both individually and collectively) control the program and what it does for them. When users don't control the program, we call it a "nonfree" or "proprietary" program. The nonfree program controls the users, and the developer controls the program, this makes the program an instrument of unjust power.

A program is free software if the program's users have the four essential freedoms:

• The freedom to run the program as you wish, for any purpose (freedom 0).
• The freedom to study how the program works, and change it so it does your computing as you wish (freedom 1). Access to the source code is a precondition for this.
• The freedom to redistribute copies so you can help your neighbor (freedom 2).
• The freedom to distribute copies of your modified versions to others (freedom 3). By doing this you can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this.

A program is free software if it gives users adequately all of these freedoms. Otherwise, it is nonfree. While we can distinguish various nonfree distribution schemes in terms of how far they fall short of being free, we consider them all equally unethical.
Open Source

• A Philosophy

“Roughly, it means that the users have the freedom to run, copy, distribute, study, change and improve the software.”
Open Source

• A Standard for Evaluating Software Licenses

The Open Source Definition (Annotated)

Version 1.9

The indented, italicized sections below appear as annotations to the Open Source Definition (OSD) and are not a part of the OSD. A plain version of the OSD without annotations can be found here.

Introduction

Open source doesn’t just mean access to the source code. The distribution terms of open-source software must comply with the following criteria:

1. Free Redistribution

The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.

*Rationale:* By constraining the license to require free redistribution, we eliminate the temptation for licensors to throw away many long-term gains to make short-term gains. If we didn’t do this, there would be lots of pressure for cooperators to defect.
The distribution terms of open-source software must comply with the following criteria:

- **A Standard for Evaluating Software Licenses**

The Open Source Definition (Annotated)

**Version 1.9**

The indented, italicized sections below appear as annotations to the Open Source Definition (OSD) and are not a part of the OSD. A plain version of the OSD without annotations can be found here.

**Introduction**

Open source doesn't just mean access to the source code. The distribution terms of open-source software must comply with the following criteria:

1. **Free Redistribution**

   The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.

   **Rationale:** By constraining the license to require free redistribution, we eliminate the temptation for licensors to throw away many long-term gains to make short-term gains. If we didn't do this, there would be lots of pressure for cooperators to defect.
Open Source

- A Standard for Evaluating Software Licenses

“The distribution terms of open-source software must comply with the following criteria”
5. No Discrimination Against Persons or Groups
The license must not discriminate against any person or group of persons.

6. No Discrimination Against Fields of Endeavor
The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, it may not restrict the program from being used in a business, or from being used for genetic research.

10. License Must Be Technology-Neutral
No provision of the license may be predicated on any individual technology or style of interface.
Open Source

• A Set of Community Expectations
  – Collaborative development
  – Right to “fork”
  – Acceptance on technical merit
    and very likely
  – Contributors’ intellectual property won’t thwart project
    • See: newer OSI license patent grants
  – Don’t have to join or sign anything else
Focusing on one part (or no parts) may cause community rejection.
Open Source

“The source code is available!”

Philosophy

Focusing on one part (or no parts) may cause community rejection

Licensing

Expectations
Open Source

Philosophy

Focusing on one part (or no parts) may cause community rejection

Licensing

“The project is open to all!”

Expectations
Open Source

Focusing on one part (or no parts) may cause community rejection

“We are using an OSI-, or FSF-, approved license!”
Open Standards
Open Standards

• Process-Based

• Outcome-Based
Open Standards

• Process-Based
  – ANSI Essential Requirements
    • Consensus
    • Public Review
    • Comments & Changes
    • Appeal
      – http://www.ansi.org/essentialrequirements
Open Standards

• Outcome-Based
  – European Interoperability Framework
    • Consensus
    • Specification freely available
    • Non-profit organization
    • No constraints on reuse
    • Patents irrevocably available royalty-free
Open Standards

Process-Based

The Middle Ground

Outcome-Based
Open Minds
Open Minds

“As long as the source code is available, the standard’s code is ‘open’”

“The standard *must* be royalty-free, for any use”
Open Minds

• Use an OSI-approved license, unmodified
• With express patent grants to contributions
  – E.g., not BSD & MIT-type
• RAND doesn’t cover open source software
Open Minds

- Open Air Interface
Open Minding

- Open Air Interface
  - “Open Air Interface Public License”
    - Apache 2.0, modified
      - Patent grant only for “study and research purposes”
      - Otherwise, contributors commit to FRAND
  - Is interface “open,” & is license “public”?
    - “The OpenAirInterface™ Software Alliance (OSA) is a ... consortium to develop ... open source software....”
Open Minds

• Hypothetical
  – Hardware + software standard
    • Hardware: specification document
    • Software: code developed by standards body
    • RAND IPR Policy
    • Software released under BSD or MIT
      – No express patent grant, or express grants disclaimed
  – Outcome?
Open Minds – Open for Questions