



http://www.micron.com/-/media/Documents/Products/Software%20Article/SWNL_open_source_software_pt1.pdf

The Significance of the Free and Open Source Software Movement

Part 1 of a 2-Part Series on Open Source Software, Linux, and Google Android™

By **Phil March, Micron Software Product Manager**
Giuseppe Russo, Micron Software Development Manager

With the popularity of the Apple iPhone®—a platform based on a proprietary operating system—and its more recent competitors based on the open Google Android™ operating system, a lot of market discussion has taken place recently about what open source is, where the concept came from, and what it means to both phone makers and software developers if the operating system is open or closed. This is the first of two articles that discuss the significance of the open source software movement and what it means to both manufacturers and independent software vendors (ISVs). The second article in this series will cover Android and its evolution from Linux.

Comparison of Open Source and Free Software

Open source software is software that is provided to a group or community in the form of source code that can be modified. With the advent of the Internet, this model of sharing source code has created an environment in which diverse production models, communication paths, and cooperation of interactive communities can thrive. While this model has had various names in recent history, the term “open source” is now used predominantly to describe production and development practices that provide access to the end product’s source materials.

In regard to computer science, open source software is software whose source code is published and made available to the public. This enables anyone to copy, modify, and redistribute the source code without paying royalties or fees. Open source software is often developed in a public, collaborative manner and evolves through community cooperation. These communities are composed of both individual programmers and large companies.

Table 1. Examples of Open Source Software

Software	Description
7-Zip	File archiving tool.
Mozilla Firefox®	Web browser.
Linux	Family of Unix®-like operating systems.
Google Android	Embedded mobile device version of the Linux operating system.
Symbian™	Full-featured mobile operating system.

Despite a substantial difference between the terms, “open source” is often confused with “free software.” To make things worse, there are different definitions of the term “free software.” According to the Free Software Definition (FSD)¹, the difference relates more to

¹ The FSD was written by Richard Stallman and published by the Free Software Foundation (FSF).

freedom of choice rather than freedom from cost. The term “free” is used in the sense of free speech, rather than cost-free.

The definition of “open source” comes from the Open Source Initiative (OSI), which is an organization dedicated to promoting open source software. One of their primary goals is to determine whether a software license can be considered open source. There are four basic freedoms in the FSF’s definition, which are described in Table 2.

Table 2. FSF’s Four Basic Freedoms²

Freedom	Description
Freedom 0	The freedom to run the program for any purpose.
Freedom 1	The freedom to study how the program works and change it to make it do what you wish.
Freedom 2	The freedom to redistribute copies so you can help your neighbor.
Freedom 3	The freedom to improve the program and release your improvements (and modified versions in general) to the public, so that the whole community benefits.

According to the definition, free software differs primarily from open source software in the category of derivative works. In a free software model the license must allow not only the creation of derivative works, but also that they be allowed to be distributed under the same terms as the original software. This concept is called “copyleft” (in contrast to copyright). In a copyleft scenario, not only is the software free to be used by others, those derivative works by others must also be made public. The intent is to ensure continued freedom and cooperation in the community. As GNU project founder Richard Stallman describes, “The simplest way to make a program free software is to put it in the public domain, uncopyrighted. This allows people to share the program and their improvements, if they are so minded. But it also allows uncooperative people to convert the program into proprietary software.”^{3a}

In essence, by choosing one term over the other one lets others know what their goals are. As Stallman states, “Open source is a development methodology; free software is a social movement.”^{4a}

Free Software is Open Software (But Not Vice Versa)

Though there are philosophical differences between the use of open source and free software, practically speaking, free and open source software refers to the same licenses. While stressing the philosophical differences, the FSF contends:

“The term ‘open source’ software is used by some people to mean more or less the same category as free software. It is not exactly the same class of software: they accept some licenses that we consider too restrictive, and there are free software licenses they have not accepted. However, the differences in extension of the category are small: nearly all free software is open source, and nearly all open source software is free.”^{5a}

² Source: <http://www.gnu.org/philosophy/free-sw.html>

³ Source: <http://www.gnu.org/copyleft/>

⁴ Source: http://en.wikipedia.org/wiki/Open-source_software

⁵ Source: Free Software Foundation, <http://www.gnu.org/philosophy/categories.html>

Examples of Free and Open Licenses and Organizations

- Operating systems:
 - AROS Research Operating System
 - Berkeley Software Distribution (BSD®)
 - Darwin
 - eCos®
 - FreeDOS™
 - GNU
 - Haiku®
 - Inferno®
 - Linux
 - Mach
 - MINIX
 - Symbian
- Organizations:
 - Apache Software Foundation
 - Eclipse Foundation
 - Blender Foundation
 - Open Source Initiative
 - Software Freedom Conservancy
 - Symbian Foundation

Linux as Example of Open Source Software

On our way to discussing the Android operating system, we must stop momentarily to discuss the open source software upon which Android was built: Linux.

Linux is a classic example of open source software. It was created in 1991 by Finnish computer science student Linus Torvalds as a personal project; the goal of which was to create a new operating system kernel. While Linux began as a few C files, it has grown rapidly and consistently over the last 20 years, both in adoption and in sheer code size. Currently, Linux includes about 370MB of source code, dozens of Linux versions, and thousands of derivative works. To date, there have been six major cycles of development and code released, with the latest (version 3.0) being released to the public on July 21, 2011.

The development of new Linux versions occurs due to work done in the community and by work done by Torvalds himself. Typically, all changes made by developers worldwide are integrated by Torvalds, along with security additions and bug fixes. A new version is typically released approximately every three months.



Micron's Involvement in the Open Source Community

Micron participates in using open source software and resubmits derivative works back to the community. This participation includes the Advanced XIP File System (AXFS) and Micron® RealSSD™ P320 solid state drive (SSD) Linux driver code, which we describe in the next sections.

Advanced XIP File System (AXFS)

AXFS is an example of an open source Linux project championed by Micron. This file system takes advantage of the Linux execute-in-place (XIP) feature, which enables faster code booting by executing straight from Flash memory and bypassing the normal step in which the code is brought into RAM then executed from there. It is primarily intended for use with NOR in embedded systems.

The project was initially registered on Sourceforge.net (a host for open source projects) on March 31, 2006. A couple years later, the first version was submitted in a request for comment (RFC) on August 20, 2008. In his posting, Micron software engineer Jared Hulbert discusses the ultimate advantages of this file system as being "smaller memory sizes and faster launches."

Following this posting, a comment thread was started with an article by Jonathan Corbet on August 26, 2008, that discusses the benefits of a file system that takes advantage of the XIP mechanism already built into Linux.

P320 Solid State Drive (SSD) Linux Driver Code

A more recent example of Micron's open source participation is the P320 Linux driver. The P320 is an SSD that is exclusively made up of Flash memory and does not have a rotating media. This SSD is based on the Linux operating system. The initial submission for this driver was made by Micron's Asai Thambi Samymuthu Patrayasamy with errata on July 26, 2011. In his code post, Thambi comments:

"This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation...This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY...see the GNU General Public License for more details."⁶

Micron system software engineer Sam Bradshaw followed up the next month with a patch to the driver. Then, on September 21, 2011, the community received a contribution back from Dan Carpenter of Oracle.

In the above case, as in all Linux cases, activities like postings, comments, and patches were communicated to the entire community through the Linux-kernel mailing list.

⁶ http://www.possamer-threads.com/lists/linux/kernel/1421632?do=post_view_threaded#1421632

To Be Continued

The second part of this series will discuss details about the Android operating system and how it fits into the open source discussion.

About the Authors



Bringing over 20 years of software product management experience to Micron, **Phil March** is a Micron Software Product Manager specializing in NAND, DRAM memory optimization and RTOS and Symbian operating systems. March has been instrumental in Micron software projects for device drivers, Symbian support, eMMC™, wireless/mobile phones, NAND Flash Translation Layer (NFTL), Sector-based Compact File System (SCFS), and Technology for Memory Optimization (TMO). Prior to his work at Micron, March spent the early years of his career as a staff scientist working on rocket and satellite upper-atmospheric environments and radiation hardening.



Micron software development manger **Giuseppe Russo** is responsible for software development, customer support, and memory optimization for both embedded and wireless customers. With over 10 years experience in the semiconductor industry, Giuseppe specializes in file system development and software optimization for storage media devices on generic RTOS, Linux, and Google Android operating systems.