License Management: How Developers Control Software Licensing

By Richard Mirabella

Editor's Note: *The following is reprinted with permission from GLOBEtrotter's Web site at www.globetrotter.com and gives a good overview of license management and licensing issues facing developers and resellers today.*

When the United States of America ratified the U.S. Constitution in 1788, Congress received the power "to promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries." Upon this legal foundation rest the rights of U.S. companies to buy and sell software. What is generally known as "selling software" is really selling a limited license to use and copy the software. I apologize for this nationalistic point of view to my non-U.S. readers. Some software includes patented technology. However, most software has only copyright protection.

The first concepts of software licensing developed in the days of non-networked computing and million dollar "low-cost" computers. These concepts included:

- licenses are fixed to a particular CPU
- prices of software licenses increase as the computer's performance increase
- limiting the number of copies of software made to disk or tape is important

When PCs became popular in the early 1980s, PC software vendors kept this earlier view of licensing software--with the exception of pricing based on CPU performance. The rapid evolution and market acceptance of networked computing in the late 1980s forced a rethinking of what software licensing means on a network of workstations, diskless workstations, servers, X-terminals and ASCII terminals.

Honest users want to live within the terms they purchased their licenses under. The old concept of software licenses allowed system administrators to keep track of software by limiting the copies of software to hard disks that were directly attached to licensed CPUs--networked file systems made this view of licenses obsolete. Workstation customers called for new concepts of software licensing:

- software is a network resource--licenses "float" on the network
- software costs are a function of how many users simultaneously run the software--pricing is based on users, not CPU performance
- value exists in the use of software, not in the number of copies on disk or tape

The new concepts of licensing require dynamic tracking of users and software licenses. The technology that does this is called a "license manager." Without license management technology, it's impractical for a system administrator with a network of even 10 workstations to keep track of how many floating licenses are used concurrently.

License management differs from software "copy protection" used in the PC marketplace by controlling execution instead of copying. PC users rebelled against copy protection because the practice inconveniences honest customers while giving them nothing in return. In contrast, UNIX customers pressure software developers to provide them with license management because it makes software a network resource and makes software pricing more equitable.

Buying more licenses is easier under license management because it eliminates the need to send out another tape with the software. All that the user requires is a "license key"--a brief string of data that describes the license plus some security codes. These license keys can be distributed by e-mail, FAX or verbally over the telephone. Today, most major vendors of workstation software use license managers to control their products.

Licensing Policies

License managers embody the licensing terms of a developer into software and datafiles. To understand license managers, it is critical to understand the types of licensing policies used by software developers.

Node-locking, once the most popular form of licensing, parallels the view that software is licensed to a particular computer. Typically, users may remotely login to the licensed computer. This licensing model is typically found in computationally intense applications and with software used on workstations dedicated to a particular application.

User-based licensing assigns licenses to a specific user identification. This is useful for products that are user dependent, e.g., an e-mail product, or business transaction applications--applications in which "lending" someone a user identification conflicts with the very nature of the application.

Site licensing software to a geographical site is another approach, one that generally has a high entry price for the customer. Both the software vendor and customer frequently spend a great deal of time negotiating over price, as site licenses generally don't match pricing to actual usage very well. Defining a "site" can also be difficult. For example, a company with a large corporate campus may be so large that multiple sites are defined. Site licenses are most appropriate when companies standardize on a specific product.

"Network licensing" and "floating licenses" are almost synonymous with license management. Floating licenses fit very well with the concepts of networked computing. Floating licenses became the prevailing workstation licensing policy due to its efficient matching of usage to the number of licenses sold. Many customers at a site can have access to a software product, but the cost of this access may be the price of one license. As the software becomes widely used, additional licenses are purchased.

Other licensing policies include specifying an "expiration date" or a "start date" for licenses, used in software evaluations or leasing programs. There is no "right" licensing policy for all products. Developers decide policies based upon the way the software product is designed and used, and on competitive factors. Sometimes the same software may be licensed with two very different licensing models. For example, an electronic publishing package may be available as either a floating license or a node-locked license, with the node-locked license having a lower price.

License management helps users choose the right software, when software vendors implement extensive customer evaluation programs. This allows users to evaluate a software product for a few weeks before it is purchased. Users at a site tend to "standardize" on a particular software product that is license managed software because the product is a networked resource that can be used by all, not just those who have a license for their workstation. System administrators also have tools to control the use of licenses. For example, administrators may make licenses available only to specific groups, or specific users.

General Design Goals

The general design goals of a practical license manager may at first seem simple. However, because license management becomes an essential service on a network, such goals can become quite complex. License managers must be very robust. Backup license servers are essential. If the license server fails, users cannot run the software they purchased. That is not a graceful failure. High availability needs to be designed-in to accommodate network failures as well. Finally, as servers go down or networks fail, the license manager must automatically configure itself without system administrators taking action--their hands are already full getting the server or network working again.

The license manager must be flexible enough to allow the product to be licensed in a way that makes sense to customers and software vendors. The license manager must understand the issues presented by X-terminals, multiprocessor computers, products that can be licensed as floating or node-locked licenses.

Although the concept of licensing by "user" is well accepted, there is not as much consistency about what is a license "use." For example, only some software licenses allow a person to run more than one "copy" of the software on the same display on the same workstation with the same user id. The "right" answer varies with the product being licensed. The license manager must allow for different interpretations of what is a "use" of a license.

Most networks are heterogeneous, supporting workstations from Sun, HP, IBM, DEC, Silicon Graphics, etc. Moreover, many applications run on several different workstations. The license manager should allow licenses for Sun applications to be shared with the HP and IBM workstations--if the software developer so chooses. Most workstations today are on large networks, so license managers may be called upon to serve hundreds or thousands of licenses at one time. The license manager's client-server processing must not place a high burden on the network by scaling linearly and allowing multiple license servers.

In-House Developed License Managers

Some of the first license managers were special purpose license managers written by software developers for their own products. Frame Technology, of San Jose, Calif., is one such notable developer.

Today, however, software developers rarely write their own license manager. Instead, they license commercially sold license managers, such as FLEX*lm* (a contraction of FLEXible License Manager) from GLOBE*trotter* Software, San Jose, California, U.S.A.

Since license management is widely used today, most system administrators prefer to learn and use only one license manager for controlling applications at their sites, rather than learning a different one for each licensed product. Even developers with their own license manager switch to FLEX*lm* because of requests from their customers for them to be consistent with the commercial license managers, the high costs of porting and supporting heterogeneous networks, and the need to provide very high reliability.

SunNet License

SunNet License was introduced by Sun Microsystems in 1990, but it received little market acceptance by software developers because it was really only supported well on Suns. Most software developers need to sell their products on several UNIX platforms, and SunNet License did not provide them with the right solution. Sun no longer actively promotes SunNet License.

SunPro, SunSoft and SunSelect have all licensed FLEX*lm* from GLOBE*trotter* Software in 1992 and have already introduced products using FLEX*lm* for license management--for example, the compilers from SunPro.

FLEXlm

Globe*trotter* Software, of Cupertino California, and Highland Software jointly developed and own FLEX*lm* (which is also known as the Highland License Manager). Globe*trotter* was responsible for development and Highland Software was responsible for sales and marketing. Since its introduction in 1989, FLEX*lm* became the de facto UNIX industry standard for license management. Most software vendors who use license management, including more than 600 of the largest UNIX software developers, licensed FLEXlm. GLOBE*trotter* estimates that FLEX*lm* has more than an 80% share of this market. In addition, over 11,000 products use FLEX*lm* today. In early 1994, GLOBE*trotter* Software purchased Highland Software's marketing and sales rights. GLOBE*trotter* now is solely responsible for the development, support, marketing and sales of FLEX*lm*.

FLEX*lm* became the dominant license manager due to its flexibility in creating licensing policies, its very robust technology of redundant servers, and its extensive heterogeneous support for UNIX platforms. Finally, as FLEX*lm* became widely known by system administrators, they encouraged software vendors to use it so system administrators wouldn't need to learn and support multiple license managers. In addition to Sun, computer companies that have licensed FLEX*lm* include IBM, Fujitsu, NEC, DEC, HP, Silicon Graphics, Siemans Nixdorf, ICL, Amdahl, Convex, Cray Research, Data General, Motorola, NCR Corp, Samsung and UNISYS.

Industry Standards

Although FLEX*lm* became a de facto standard through its high acceptance rate by major software vendors and users, other companies wanting to sell license managers propose new APIs as "standards" that differ from FLEX*lm*.

In 1989, Apollo Computer, Highland Software and others formed an ad hoc group called the Software License Working Group. This group led by Apollo, developed an API for license management largely modeled after Apollo's NetLS. FLEX*lm* and NetLS both supported this API, but software developers didn't show any interest in this "standard."

In 1991, SPARC International, an organization funded by the major vendors of SPARC technology, put license management into its SPARC Compliant Definition. The purpose of the SCD definitions is to ensure hardware vendors and software vendors that any SCD application (know as SPARCware™) can run on any SCD system. Parts of FLEX*lm*'s technology are part of SCD 2.0 through the License Information Service (LIS) ABI Option. SPARC International defines an Option as an interface that is not required to be present in a conforming system. However, if it is present it must be present in full and in exactly the manner described in the SCD. If it is not present then a conforming system may not use the names described in the specification for the option.

While not a standard body because it tries to make products, the Open Software Foundation (OSF), tried to create new "standards" for license management in 1991. In the case of OSF's DME (Distributed Management Environment) Highland Software chose not to submit FLEX*lm*'s technology to OSF's DME Request for Technology due to business issues that would arise from working with OSF. DME is widely viewed as having failed. As of September 1994, there were no licensees for DME.

In 1992, another ad hoc group of software vendors that includes Microsoft, Novell, Highland Software, HP, DEC and others have proposed another API for license management. This specification covers a limited set of client APIs, and is known as the License Service API (LSAPI). GLOBE*trotter* put this LSAPI into FLEX*lm* in Q1 of 1994, but to GLOBE*trotter*'s best understanding, no software vendor is actually using this API. Very few applications are using this API (probably less than 10). Currently X/Open is discussing license management in its white papers and may consider this area for the creation of an X/Open API for license management. No specific APIs are currently being considered.

There are now more different proposed license management "standard" APIs than there are major vendors of license management technology. This proliferation of proposed license management APIs has hindered efforts towards having an official standard for software developers to use. As anyone who has followed the process of standards organizations knows, proposals by groups don't decide what are true standards. The reality of what developers choose that determines standards.

Richard Mirabella is the Vice President of Sales and Marketing at GLOBEtrotter Software and has been involved in the introduction of license management technology to the software industry since 1990.