Word Power: a Glossary of Software Licensing Terms

Save to myBoK

This Software license agreement is entered into by and between Sample Software, Inc, a Delaware corporation with principal offices located at 100 Maple Lane, Chicago, Illinois 60601 ("Sample") and _____, a corporation with principal offices located at _____ ("Licensee") and shall become effective on _____ the date it is signed by Sample ("Effective Date").

What follows this statement can be a long and bumpy read through legalese and tech talk. Defining the common terms that appear in software licenses can make reading a EULA-see below-a little smoother.

Application. Usually an entire set of software programs that provide a particular functionality. For instance, a dictation system may actually be a compilation of various software programs designed to work together.

Client. An application that invokes, typically via a network protocol, the software functions provided by one or more servers. A client or workstation usually is a computer that does not provide any network services or store any of the application files locally. Typically, this is a personal or desktop computer with access to a server via a network connection. (See also thin client)

End User License Agreement (EULA). A legal contract between a software vendor and the user of that application. The EULA, often referred to as the "software license," is similar to a rental agreement-the user agrees to pay for the privilege of using the software and promises to comply with all restrictions stated in the agreement.

Escrow. Typically an arrangement in which a trusted third party holds the source code of an application to ensure that the code will remain available to the user should the vendor cease supporting the application (or fail to meet other agreed-upon conditions).

Functionality. The intended purpose of an application being licensed; that is, the actions the software programs are described to perform in a request for proposal or marketing materials. New functionality is often sold or licensed separately as a functional enhancement (as compared to a software "update," designed to maintain existing functionality).

Integration. Linking two or more application programs to share functionality and controls.

Integration testing. Testing in which software or hardware components are combined and tested to confirm that they interact according to their requirements. Integration testing can continue progressively until the entire system has been integrated.

Interface. The linking of two or more systems to facilitate data interchange between the systems.

Interface engine. An independent application that processes interfaces between two or more other, disparate computer systems by acting as an intermediary data conversion tool.

Licensee. The individual or organization entering into a contract with another individual or company to license the use of software that belongs to the second individual or organization.

Licensee grant. A standard component of software contracts that dictates limitations on transferability and distribution of software (use by subsidiaries and affiliates), where software can be used (physical locations), who is a legitimate user, and the purposes for which the software can be used.

Licensor. The individual or company providing the software application to be licensed (often referred to as the "vendor").

Messages. Formatted data from an originating system that may be sent through an interface to another system.

Modification. Typically, the changes made to the vendor's standard system to accommodate the particular needs of the individual or organization licensing the system.

Object code. The "compiled" source code or program that typically provides all or a portion of the functionality of the application. Technically, this is the "machine code" generated by a source-code language processor, such as an assembler or compiler, which can be run as a user program.

Per-seat licensing. An agreement in which every computer that will access the application requires a license and each properly licensed PC, workstation, or client can then access the application servers on the network. Client-access license (CAL) is Microsoft's term for the client licenses needed to access a Microsoft-based server.

Per-server licensing (concurrent user licensing). An agreement in which a license applies to a particular server and allows for an additional concurrent connection to that server. The licensee must have at least as many licenses dedicated to that server as the maximum number of clients that will connect to that server at any point in time. This type of licensing is often referred to as concurrent user licensing. While this type of licensing may be more efficient (fewer unused licenses at any given point in time), once the maximum number of simultaneous users has been reached, all subsequent users seeking to gain access to the server will be denied until another user logs off or the number of licenses is increased.

Reverse engineer. The process of working backward from a vendor's finished product to discover its technological foundation. The practice is typically forbidden in contracts- along with decompiling or disassembling the software-to prevent the creation of unauthorized or unlicensed products built on the vendor's proprietary logic.

Server (file server). The computer that stores application and data files for all workstations and clients on a network.

Shrink-wrapped and click-wrapped software. Noncustomized software purchased "out of the box" or "off the shelf" and bought on shrink-wrapped media (e.g., computer discs). When downloaded from the Internet, it is often referred to as "click-wrapped."

Source code. The actual program in programming language (e.g., C++, Basic, Pascal), which must then be "compiled" into object code and run by an end user. An application may consist of many programs written in various languages that work together to provide the intended functionality. Source code is generally not provided to the licensee as part of a contract.

System or application testing. Testing to confirm that all code modules work as specified and that the system as a whole performs adequately on the computer hardware platform on which it will be deployed.

Term. The amount of time the software can be legally used by the licensee. Generally software is licensed and priced by the month or the year. Though less common, a perpetual license with indefinite renewal period is legally acceptable in the US.

Thin client. A simple client program or hardware device that relies on the server to deliver most of the system's functionality. Thin clients typically still require client licensing for the application as well as licenses for the thin client software (e.g., Citrix, Winframe, Terminal Services CAL).

Third-party licenses. Often applications will incorporate software from other vendors within their own application (e.g., database and report-writing software). These applications may require additional licenses for the server(s) or the clients and be covered under separate agreements or contract addendums.

Unit testing. Testing where a developer (usually the one who wrote the code) proves that a code module (the software "unit") meets its requirements.

Update. Changes typically made to an application to maintain its performance, fix glitches, or move to a newer format (e.g., move the application onto a new database format). Generally, updates are covered in software support.

Upgrade. Changes typically made to an application to add new functionality. Upgrades typically require new licenses.

User acceptance testing. Testing in which monitored users determine whether a system meets all their requirements and whether it will support the business for which it was designed.

Warranty. The period of time in which the vendor or licensor insures that the application will perform as described.

Source

AHIMA. "e-HIM -Conversational IT." Online continuing education class. Available online at http://campus.ahima.org.

Article citation:
AHIMA. "Word Power: a Glossary of Software Licensing Terms" *Journal of AHIMA* 77, no.4 (April 2006): 42,44.

Driving the Power of Knowledge