

John Tomeny
2007 SAM Practitioner of the Year
Convener, ISO/IEC 19770-3
Software Entitlement Tag standard
603-643-3351
johnt@sassafras.com

Sassafras Software is an early pioneer in the development of software license management technology for desktop computers.

Sassafras has been providing IT asset management solutions since their first public release of KeyServer in 1990.

K2 is an integrated software asset management (SAM) solution that enables IT managers to automatically audit, track, and control their software and hardware assets.

Asset-centric vs. Entitlement-centric Software Asset Management (SAM)

The software industry has always struggled with two conflicting streams of thought on software licensing. At the risk of oversimplification, on one side of the table sit software publishers whose primary interests are intellectual property protection, piracy prevention, and revenue generation. On the other side are SAM practitioners who simply want to get useful software into the hands of users. To do this, they need appropriate, affordable entitlements which can be easily managed.

For now, I will label them respectively "Asset centric" and "Entitlement centric", but further clarification on these labels is in order later. Under the asset centric model, nearly everything is driven by file existence - piracy prevention and entitlement compliance being the key goals. SAM practitioners are stuck counting copies of software. While under the entitlement centric model, everything is driven by... well, entitling users to use.

From a license compliance perspective, when an SAM practitioner is focused on entitlements, they can be set free from concern about file existence. But alas, only a few common software licenses today are completely consistent with an entitlement centric focus. Concurrent use licensing ("CUL") and Client Access Licensing ("CAL") are two examples that focus on entitlement without concern for file existence.

But the impossibility of managing CUL on mobile computers when off-network, along with the difficulty of buyers and sellers converging on acceptable pricing plans have discouraged its popularity. CAL licensing suffers from the lack of convenient management tools for allocating the entitlements. Also, by tradition, CAL has been applied only to remote services even though it is equally applicable to software installed or running locally. Fortunately, both CUL and CAL licensing models can be adapted for use in desktop, virtual, and mobile computing when managed by an appropriate license management tool. More on that later.

Compliance management becomes unwieldy when it is defined by file existence rather than by licensed usage. SAM practitioners, if given the opportunity to manage entitlements and compliance without being required to track file existence, will take the entitlements-focused approach every time. Software publishers can build market advantage by licensing their products in ways that make them easier to manage.

The most common licensing models today rarely provide the perfect fit for both publisher and customer requirements. Here are some suggestions that you can offer to your favorite publisher the next time you are at the negotiating table.

- 1) Single computer (node-locked) licensing in a world of virtual computers and virtualized applications gives SAM practitioners a headache akin to herding cats.
- 2) Concurrent-use licensing is often a poor reflection of value and is not easily managed with mobile computing workforces.

K2 KeyAuditor & KeyServer

K2 is an integrated software asset management (SAM) solution that enables IT managers to automatically audit, track, and control their software and hardware assets. Sassafras Software has provided software license management technology since 1990 to some of the world's largest corporate, government, and educational enterprise networks.

In addition to comprehensive hardware/software auditing, K2 will also control access to licensed software and run detailed deployment and usage reports for an enterprise-level organization or for specific departments.

New in Version 6.1:

- *K2 Web Reports: Scheduled Reports and Web Console*
- *Rules-based Filtering for Hands-free Management*
- *Role- and Scope-based Administration*
- *Enhanced Integration with Active Directory*
- *Rapid Client Deployment with MSI and MPKG files.*
- *Universal Binary Support and Dashboard Widget.*

Supported Host & Clients:

Windows, Macintosh, NetWare, Linux, BSD Unix.

More information:

*www.sassafras.com
603-643-3351*

- 3) Site licensing often counts the wrong objects for licensing purposes (numbers of computers or numbers of people, rather than numbers of computers that will use the software).

The maturation of the SAM industry is putting pressure on these conventional licensing models. The industry needs new models that are better aligned to enterprise business processes and IT configuration management. In a paper that I presented at IAITAM's 2007 Annual Conference (available on request), I explained the weaknesses of these common licensing models and pointed to alternative licensing models that have been living in the shadows but could emerge to serve enterprise computing demands today.

Let's look at some of those demands and how the new models might fit. In broad terms, organizations of any size with managed computers need to deploy software and demonstrate compliance on a mix of computers including desktops, thin clients, virtual computers, and mobile computers.

Any of the conventional models will work well in a desktop-only environment. Node-locked licenses, however, don't work well for virtual computers which are designed to appear frequently across numerous physical computers, and may disappear never to be seen again. Concurrent-use licenses can not be correctly managed on mobile computers that are often away from the network. And site licenses are often difficult to negotiate due to poor information on user requirements and hence value.

But consider for a moment a license that is defined essentially as a blend of traditional node-locked licensing and concurrent-use licensing, where the entitlement has a specified "time-to-live" along with an automatic renewal policy. For sake of discussion let's call this a TTL license. Think of TTL licenses being managed by a license server in much the same way that IP addresses are managed by a DHCP server.

When a computer connects to the network, the DHCP server assigns or "leases out" an IP address with a specified lease renewal time (a.k.a. time-to-live). Assuming the time-to-live is one week, the computer will retain the assigned IP address so long as it uses the address at least once a week. But if the computer goes away from the network for longer than the lease period (one week) it may lose its address to another computer.

Under TTL licensing, instead of allocating or renewing an IP address upon connection to the network, a software license is allocated or renewed upon execution of the software. Now suppose the time-to-live for this license is set for one month. It becomes easy to see how management of licensing under such a model could work exceptionally well to support desktop, virtual, and mobile computers in IT environments that are continuously replacing and upgrading computers.

Almost no effort needs to be devoted to license management in this scenario since the central license manager does the work automatically. A user launches a program and is granted a license. As the user continues to use the software month after month, the license is locked to that computer. But when the computer is retired, or if it is reassigned to another user who does not need that software, the TTL license expires automatically, and in time, returns to the central pool to be used by another.

As we explore this idea of DHCP-style licensing, we can see how a software publisher could adjust the TTL timer to create a CUL or node-locked license. A TTL period that is set to one second equals CUL, while a TTL period set to five years produces a node-locked license. Thus, any licensing entitlement on this continuum can be managed by the same central license server.

For another example of an entitlement based licensing model, recall the ill-fated "Adobe License Manager" introduced in 2006. Despite difficulties with implementation, Adobe introduced a brilliant software licensing model. Again, for discussion, let's call it an Activation-based license. This was not an entirely



new concept, but it was likely the first time that this model had seen such broad distribution.

Under ALM, their Activation-based license was granted to an installed copy of software when the software was given its serial number - or activation code. A central license server granted the software the ability to launch, and that license grant stayed with the software for as long as the user wished. If at some time the user or licensing administrator wished to 'harvest' the license to redeploy it elsewhere, it was a simple task.

The uniqueness to Adobe's approach is that the publisher did not require the customer to uninstall the software if it was not licensed. File existence didn't matter because a central licensing authority could be used to either assign or remove launch privileges. The net result was that IT configuration managers could clone disk images to thousands of computers and then select a subset of computers that they wished to activate using Adobe's entitlement-centric SAM tool.

One more example. I promised to give more information on how CAL licensing can be adapted for use in desktop, virtual, and mobile computing. The traditional model of a CAL (Client Access License) is to grant a client computer access to run a program or process hosted on a remote server. But why not use the model to grant a client computer access to software without regard to where the software is installed or executed - even locally installed on the client, or in a virtualized environment?

In a sense, this is just another way of illustrating Adobe's approach: activation-based licensing. But this model, unlike Adobe's implementation, demonstrates how it is not limited to managing node-locked licenses. So long as there is a central licensing process in place, virtually any type of license can be managed without the ambiguous issue of file existence.

As virtualized environments continue to expand, the limitations of audit centric SAM will become more obvious. The many forms of virtualization blur the distinction between remote versus locally installed software and remote versus local execution. In such an environment, an entitlement centric approach to licensing is really the only option.

When I first mentioned "Asset centric" vs. "Entitlement centric" licensing, I promised to provide further clarification later. Ultimately, software license compliance is always entitlement centric. In truth, the real "asset" is the license. The software is merely a resource that is provided with the purchase of the license. As an industry, we must then shift our focus to building ways in which we can deploy and manage entitlements for software usage rather than trying to manage the location of the software itself.

Sassafras Software is an enthusiastic participant in the active standards development work of ISO/IEC 19770-2 and 19770-3 which we hope will facilitate this evolution toward entitlement centric licensing. In the meantime, as the industry awaits the outcome of the standards efforts, Sassafras Software will continue its tradition of introducing new technologies and licensing paradigms that will permit the SAM and software industries to move closer to these ideals.

John Tomeny of Sassafras Software Inc is a sixteen-year veteran of the Software Asset Management industry, the Convener of the ISO/IEC 19770-3 working group on Software Entitlement Tags, a recipient of the 2007 SAM Practitioner of the Year award, a member of the International Association of IT Asset Managers (IAITAM), a member of the IT Service Management Forum (itSMF), and a founding member of the International Business Software Manager's Association (IBSMA).

Portions of this article were first published in IAITAM's ITAK Magazine in October 2008. <http://www.iaitam.org/ITAK.htm>

Copyright © Sassafras Software Inc. All rights reserved.