SYSTEM FOR ADJUSTABLE DIGITAL LICENSING OVER TIME
(THE THROTTLING PATENT)

Inventor: Ric B Richardson
Assignee: Uniloc Limited (Singapore)

I, Ric B Richardson of Huntington Beach, CA have invented a licensing system for digital products that allows the license terms over time to allow license terms to be adjusted around the target customer's usual usage patterns but invokes license restriction policies when customers intentionally abuse license terms.

BACKGROUND

A common capability of digital product license systems is the ability to control how many devices are allowed to be used with each product license which is usually sold to an individual customer or company. For example US patent 5,490,216 refers to a system where a license is given to an individual, but in turn that license is linked to a specific personal computer thereby limiting the copyright holders exposure to copyright abuse if the user decided to share their license with other unauthorized users.

A problem that has arisen over time is the fact that consumers of software have normal patterns of use that include the installation and use of digital products on multiple devices. For example a person may wish to buy software and use it on three computers at their home, a computer at work, a mobile computer and the computers at their holiday home and their parent's house. In addition to these uses, computers are also bought, sold and replaced so over time maybe two or three times this number of computers may be used by the user over time with a legitimate need to install and use the software on every computer.

Publishers of digital products have a dilemma in that they may want their customers to receive the normal freedom to use the software that they have purchased but they also do not want the software licenses to be freely shared amongst end users or even in worst case shared anonymously over the Internet resulting in massive piracy and copyright abuse of the product.

To solve this problem some publishers have set a relatively high device to license ratio in their control systems in the hope that customers will not exceed the maximum number of devices per license. An example of this is Apple iTunes which enables customers to play a purchased music file on up to five PCs per license before being requested to buy an additional license. They have also implemented a system that allows customers to turn off the license rights of individual PCs with regard to a specific music file license and therefore free up that device installation so that the music file can be used on one additional PC.

While this method does go some way to appeasing the problem of a normal customers usage expectations, it does not take into consideration the normal attrition that occurs with the purchase and upgrade of PC devices and places an expectation on the user to go through a number of involved steps to retain their rights to use the software.

The proposed invention addresses this and other problems by allowing for a changing number of device installations on a per license basis over time. The invention also includes a customer feedback system that warns a user when they are nearing the limit of their device installation ceiling for their license.

An example scenario could be as follows. A software publisher wants to commence distribution of a software product and to minimize unauthorized copying of their software. Their license may state that they authorize the user to use their software on up to five devices but reserve the right to increase this limit at their own discretion. The customer installs the software on the three computers they have at home. Each time the software connects to a license management server controlled by the publisher over the Internet to ensure that the device limit for the individual license has not been exceeded.

The customer then chooses to install the same software on their PC at work. Upon contacting the publishers license management server over the Internet a message is displayed to the user warning them that they are nearing the limit of their device count for their license.

Two weeks later the user wishes to install their software on the two computers they own at the customers holiday home. If the publisher uses the proposed invention the maximum number of devices for the license may have been adjusted to accommodate a reasonable small increase in the number of devices linked to a specific license and both PCs may be allowed to install and run even though the publishers stated device limit per license is five.

Then three months later, the user experiences water damage from a flood in their house and a new PC is purchased. Upon installation of the protected software the invention will allow the user to obtain additional device installations from the publishers license management server for the same license even though the device limit is five.

However, if that user shares their license with all the computer users in a college dormitory, the invention can be set to stop wholesale abuse of the license terms.

PRIOR ART

Patents:

Uniloc patent 5,490,216 and Microsoft patent 6,243,468

Web search terms:

Throttling license/ software/ authroisation, software license, time based license, timed license, expiring license, expiring certificate, activation, timed activation.

Companies researched for prior art:

Crypkey, Apple, Macrovision, Microsoft and other small related companies.

DRAWINGS

Figure 1: example License

Figure 2: system for license and device authorization

DESCRIPTION AND OPERATION

The described example embodiment of a system for adjustable digital licensing over time allows a software user to increase the number of devices they can use with a particular software license 60 over the period of ownership of that license.

Initially, the publisher or distributor of the software sets rules 60 that govern the use of the software on a specific number of devices. These rules are described as an example embodiment in figure 1. The number of devices allowed to run the software in an authorized or enabled state increases over time to reflect the normal usage pattern of software users where the user adds devices, replaces or upgrades devices over time. The rules 60 reflect this pattern of an increasing number of devices authorized over time. For the first five days of the users use of the software a total of five devices can be authorized on new devices. For the next twenty-five days until the thirtieth day after first authorization, the user is allowed to authorize a total of seven new devices. After the first thirty days an additional four devices can be authorized, delivering the maximum number of copies on separate devices under the license which, in this example embodiment, is eleven.

Figure 2 shows an example embodiment of a software system that is designed to manage the rules terms 60 described in figure 1.

Device locked license systems such as described in US5,490,216 enable software license to be locked to a license agreement and specific authorized devices. The system usually comprises a device that requests authorization 50 by means of a software process 50 and the licensing authority 55 which is usually a software system that represents the publisher or distributors interests and regulates the number of devices that can be used with each license.

Typically the device requesting authorization 50 collects license related 10 and unique device identifying information 11, compiles it into a communication and sends it to the authorization authority 55. Upon receipt the license authority 55 checks that the license information is valid 13. If the request fails, an authorization is disallowed 14 and the device based software is sent a message to this effect. In practice this will involve further action by the device based software to notify the user of the failure to authorize and then either terminate the software or allow the software to continue in some form of trial mode.

If the request for authorization 12 includes license data that is valid, the license information checking process 13 will pass and the requesting devices unique identity information 11 is checked to see if it exists in the database or prior authorizations 15. If the device identity exists 16, meaning that the software has been successfully registered on the same device in the past, then according to the license terms for the software 60 a re-authorization is automatically allowed 17. A communication allowing the software to continue in an authorized state is passed to the requesting device software 50 and the software on the device is subsequently authorized 18 and allowed to run.

If the unique identity of the device 11 is not in the authorization database 15 of previous device requests, then the licensing authority 55 checks to see if the new authorization request is the first request or is a subsequent request that has occurred in the first five days from the date of the first successful authorization 19.

If the request is within the first five day period 19, the authorization database 15 is consulted for a count of how many successful authorizations for new devices have been allowed. Under the license rules 60, if the device count is less than five then a message is sent to the request device that allows the software to continue in an authorized state 18. If the device count is equal to five then the licensing authority 55 sends a message to the requesting device 50 allowing the device to run in an authorized state 18 but also may optionally inform the user that the limit of the number of devices available to run under this license has been reached 22 and that subsequent requests for authorization may be denied in the short term.

If the count of devices authorized for use with the specific license 10 is greater than five 23, then the licensing authority 55 sends a message denying authorization 25 and the user is optionally notified that the limit of devices that can be authorized with their license terms has been exceeded 24. In practice, the software on the requesting device 50 will either subsequently terminate the software or allow the software to run in a limited trial mode if this is available.

If the number of days since the first authorization of a device for the license 10 is not less than six 19, then the licensing authority tests the time elapsed from the first successful authorization to see if it is less than thirty-one days since the date and time of the first successful authorization 26. If this test 26 is successful then a test is made to see if the count of successful new device authorizations is less than seven 27. If this is so a communication is made to the requesting device 50 authorizing the device to run 28. If the new device count is equal to seven 29 then the user is warned that their device limit has been reached 30 and the device is subsequently authorized to run 28.

However, if the new device count is greater than seven 31, a communication is made to the requesting device 50 that the authorization is denied 33 and optionally the user is notified that their license device count has been exceeded 32.

If the number of days since the first successful authorization is greater than thirty days 34, the device count for the license 10 is checked in the authorization database 15 and the device count for the license retrieved. If the number of successful new device authorizations is less than ten 35 then the device authorization is allowed 36. If the device count is equal to eleven 37 then the user is optionally warned that they have reached the limit 38 and the device is authorized to run 36.

However if the device count is greater than eleven 39 then a communication is made to the requesting device 50 that the user be optionally notified that their device number terms of the license have been exceeded 40 and the authorization is denied 41.

The result is a license system that allows consumers of software to load their software on new or replacement devices as they are purchased over time without exposing the publisher to copying abuses that is common amongst software pirates and casual software copiers.

ALTERNATIVE EMBODIMENTS

Alternative embodiments of this invention includes but are not limited to:

A system as above where the license management system is linked to a fixed calendar date rather than the date of first successful authorization. This approach can be used for marketing and distribution purposes such as specifying specific periods of high copy counts to encourage word of mouth and user to user sharing but later restricting the device count to encourage people to begin paying for copies that have been intentionally shared.

The example embodiment is simple for the purposes of understanding but can include any number of evaluation periods, not just the five, thirty and unlimited day periods described in the example. Also the number of notification stages can be indefinitely expanded, for example the user could be given a polite message encouraging them to be careful with making copies when they are two copies away from their count limit and a stronger message when it is their last copy before being denied authorizations. Messages could also optionally tell the user how many days they have to wait before additional device authorizations will be available.

The example embodiment also shows that the allowed copy count increases over time. An alternative embodiment could be used where the allowed copy count decreases over time. This may be useful in a situation, for example, where the publisher supplies their software with a fairly open device count license rule but discovers individual instances of copy abuse and decides to limit the license terms of those specific licenses.

The described system could also be used with authorizations for software that is rented or otherwise allowed to be used for a specific period of time or number of uses rather than indefinitely as in the example embodiment.

An alternative embodiment of all of the above scenarios could include an algorithm rather than an arbitrary value in calculating both the time period for the calculation of the device count, and the device count related to that specific measured time period. For example, the algorithm for the available device count could be equal to the number of elapsed days since the first successful activation divided by five in brackets plus five. Using the example algorithm a device count of five would be available from day one, and a device count of eleven at day thirty and so on.

Another alternative embodiment could be used for security applications where access is granted to data or some other valuable or important item as a result of a successful authorization rather than in the example embodiment where it is a license that is being granted.

Example License Rules

License allows:

- 5 devices to be added within the first 5 days of the initial authorization date and time.
- 7 devices to be added within the first 30 days of the initial authorization date and time.
- 11 devices to be allowed in total.
- Indefinite numbers of re-authorizations for devices already authorized.

60

Figure 1

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.