



Why Should Developers Use License Management Software?

Yes, **software piracy** is widespread. However, combating theft of your software products is just one of many reasons for software developers to consider license management, and sometimes not the most important. The hacker gleefully installing a pirated download of your software may be a concern for game or tax software vendors, but enterprise application developers can easily reassure themselves that this thief wouldn't have bought from them anyway, so what is the loss?

For these complacent enterprise software developers, consider these scenarios:

The mobile **light-fingered employee**

A user at your legitimate customer leaves their job to move to a rival company. Before they go, slack customer security allows them to load up on your software and take it to their new employer. Result: lost revenue to you.

The '**honest abusers**'

The company signed your license agreement and purchased a number of licenses. However, once outside the control of the IT department, copies spread throughout the company and quickly exceed the number of licenses purchased. You may catch up with them later at upgrade time, or if you perform an audit, but you really don't want to be having this kind of discussion with your customer. Result: lost revenue to you.

Increase your revenues

License management can be a very powerful tool for you to **increase your market share**, better manage your revenue flows, and increase customer satisfaction. The huge win for you and your engineering department is that all this can be done without custom engineering or the proliferation of many builds of your code. Engineering integrates the license manager into the code, then hands off one version to production: marketing/sales can now use the license manager to determine how long each copy will work, what functions will work, what machine it will run on and so forth. Engineering can get on with the next release.

You can offer a **trial version** of your product to prospective users. If your product has a broad application, you could even make it available for download from your website. The license manager makes sure the trial version works for a set period, or for a limited number of high-value operations, or some functions are unavailable, or all of the above.

You can **segment your market**. Offer a low-end version with limited functionality, and an enterprise version with more features. You could also offer different versions tailored to specific application verticals. In fact, these are all the same binary with a different key! If your entry-level users want to buy another function, all you have to do is send them a new key, with no distribution cost or delays.

Manufacturers of complex hardware systems are taking advantage of this capability to dramatically lower their costs using **system versioning**. Instead of manually configuring every system to a specific customer order, simply ship a standard system (or perhaps a low, medium or high configuration system, or a system configured for a specific vertical market), then enable the functions purchased by that customer using a software key. Manufacturing, inventory, supply, distribution, order fulfillment and maintenance are dramatically simplified. And if the customer wants to upgrade their machine, you just send them a new key to turn on a capability already in their system, for very high upgrade margins, no upgrade costs, immediate delivery and a happy customer.

You can **cut your sales cycle** by offering a license model that suits how the customer wants to buy. For example, as well as offering perpetual licenses, you could offer a subscription license, where they pay every month or every quarter. The lower initial expense may be an easier commitment for the customer. It can also fall under a lower-level signature authority, and be paid out of an operating not a capital budget. Speed up the sales cycle by finding an available budget and matching the expense of your product to how they want to pay.

Once you have developed and released your latest software, wouldn't you like to know how your customers are actually using it? Some licensing systems can keep track of users' license requests in a relational database, giving you a mine of **business intelligence** on usage patterns. Once you know what your customers are doing, you have the data to drive product improvement and set engineering and marketing priorities.

Cut costs

With the ubiquity of internet connections, and people's comfort with self-service web sites, software developers have an opportunity to **improve productivity**. Instead of processing an order manually and shipping a disk or a dongle, you can take an order on your web site and instantly email the key to the user. You can even automatically re-validate licenses if customers need to move their software to a new system.

“How hard can it be?”...

As software developers you may think a license manager is something you can readily develop for yourselves. Why pay a third party for a commercial product at all? This is a typical phase in new software technologies – initially the application is simple to build and doesn't require specialist knowledge. As the breadth of functions expands, and the

technology becomes more sophisticated, it is ever harder for developers expert in other fields to build a product that can match commercial products on features or cost, until in the end no-one would consider building their own. Building an effective license manager (and how will you know if your home-grown tool is not effective?) involves more than just encryption: system clock detection, metering, machine locking, flexible licensing policies, feature licensing, composite licensing, key generation and management and so forth are all required. People used to write their own databases, and even operating systems, but the capability and value proposition of commercial solutions is now far beyond the reach of a custom developer.

License management is on the same curve. Leading license management tools offer an off-the-shelf tested solution with sophisticated security measures, continuous development to counter new hacking threats, flexible license models, built-in business processes, user interfaces and documentation for a fraction of the cost of a custom development. It just no longer makes business sense to “roll your own”.

Agilis Software
39270 Paseo Padre #228
Fremont CA 94538
USA
<http://www.agilis-sw.com>
Tel.: (510) 550 4340