

OVERCOMING THE SOFTWARE LICENSING COMPLEXITY CRISIS

The Case for a Universal Licensing Platform

Gartner.

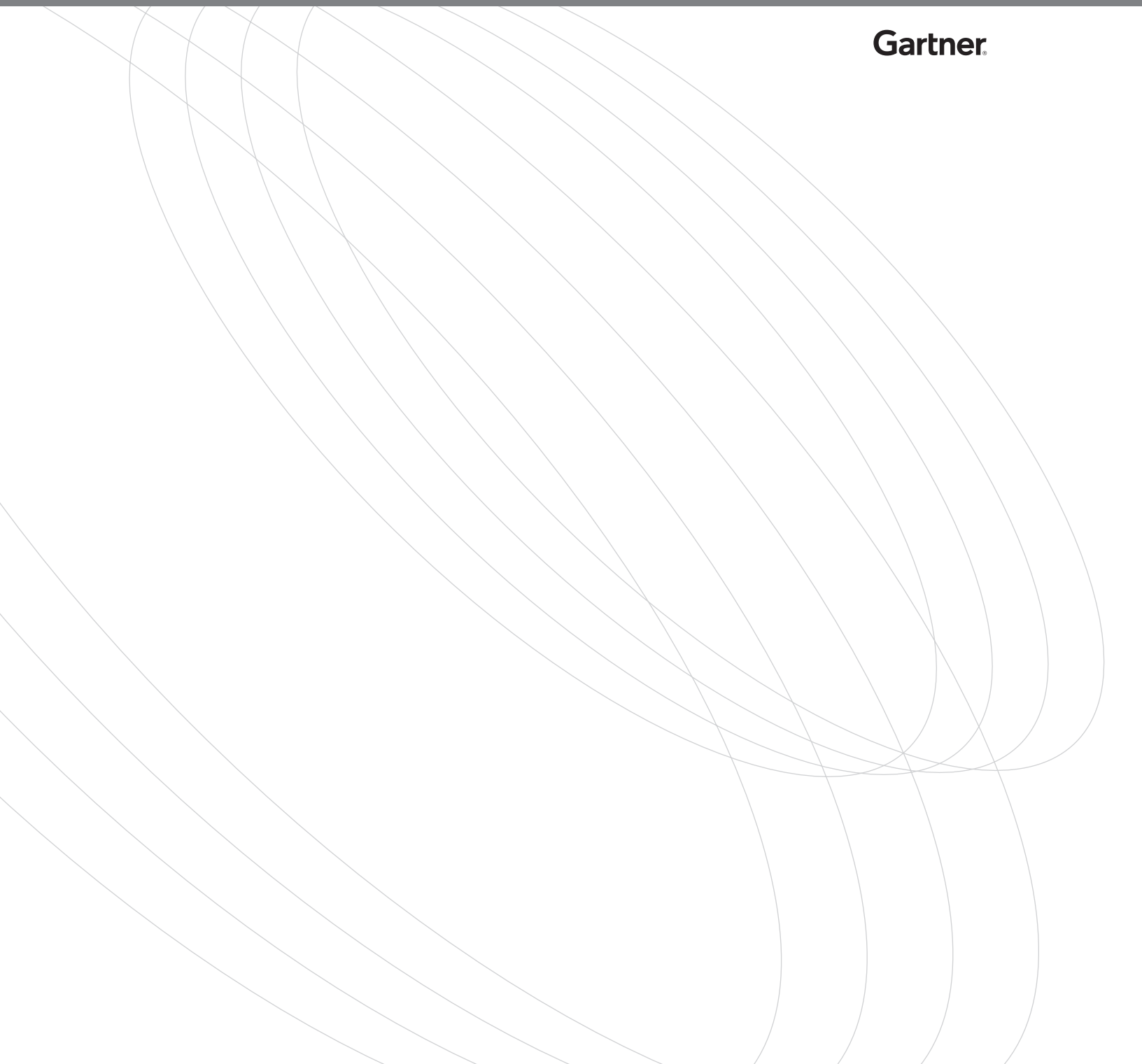
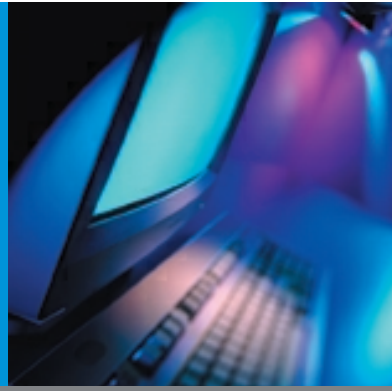


TABLE OF CONTENTS

The Case for an Integrated Licensing Platform	2
Implications of an Integrated, Modular Platform	3
Modularity » Flexibility » Responsiveness » Revenue	4
The Appeal of “Utility Pricing”	4
Management and Compliance: Publishers Get Paid When Customers Have it Easy	4
Source-Level Modularity as a Cost-Reduction Strategy	5
The Value of Being “Future-Proof”	5
Clearly Define Software License Usage Rights <i>Gartner Research Note, April 17, 2003, Jane Disbrow</i>	6
Named User License Model	6
Concurrent User License Model	7
Server-Based License Model	7
Role-Based License Model	7
Employee-Based License Model	7
Financial-Based License Model	8
Transaction-Based Pricing License Model	8
Bottom Line	8

OVERCOMING THE SOFTWARE LICENSING COMPLEXITY CRISIS

The Case for a Universal Licensing Platform

This paper presents a top-level view of the mounting difficulties plaguing software publishers in pricing, packaging and licensing their products. It explains how publishers can overcome these challenges, reduce costs, and increase revenues by moving from traditional licensing solutions to a modular, integrated licensing platform.

Software publishers are constantly searching for new ways to grow profitably while still satisfying their customers. Increasingly, they are turning to electronic licensing in hopes of streamlining the process of pricing and packaging their products while adapting rapidly to shifting market demands. Unfortunately, publishers are confronted with a growing crisis of complexity that often prevents them from achieving these goals. Some of the obstacles they face:

- Licensing systems today must interact smoothly with many other business systems and processes within a publisher's organization. This creates costly integration challenges and raises daunting issues of compatibility and interoperability.
- A publisher's various departments and business units often have procedural requirements that seem too diverse to be addressed by any single licensing product or vendor. This can result in the adoption of a variety of incompatible solutions that don't work together efficiently.
- Major software purchasers now routinely insist on individualized pricing. This forces publishers to absorb the cost of scrambling to respond with new and more flexible licensing models.
- Corporate purchasers are demanding more cost-effective ways to manage their software assets and ensure compliance with licensing terms. These efforts are further challenged by the rise in decentralized purchasing.

Figure 1 illustrates how the complexity of software management has increased exponentially over the past two decades. Complicating this environment further are the many mergers and acquisitions among publishers, leaving in their wake a hodgepodge of incompatible, and often incomprehensible, licensing systems.

KEY TRENDS IMPACTING THE SOFTWARE INDUSTRY

Software management complexity has increased exponentially

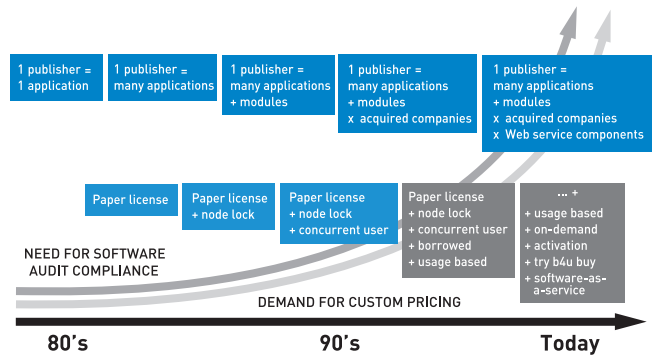


Figure 1: Key Trends Impacting the Software Industry

The Case for an Integrated Licensing Platform

Addressing the complexity of licensing is crucial to reducing and controlling costs for software publishers. The challenge is that licensing solutions cannot operate independently of other critical business processes. Indeed, licensing can have an enormous impact on every aspect of a publisher's organization (see Figure 2).

THE CASE FOR AN INTEGRATED LICENSING PLATFORM

LICENSING AFFECTS ALL OF A PUBLISHER'S DEPARTMENTS

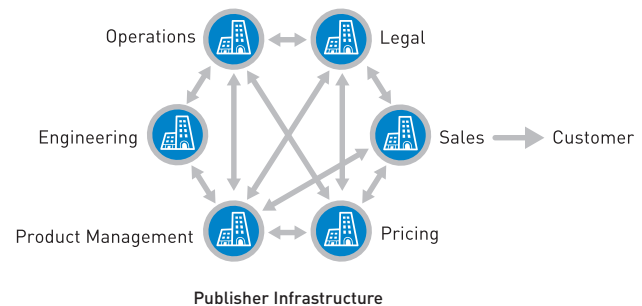


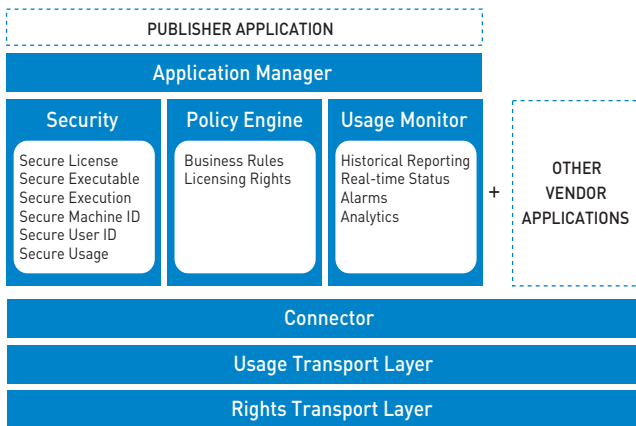
Figure 2: Licensing Impacts All Business Processes

To assure that this impact is positive, a licensing mechanism must provide well-defined interfaces to other business processes and systems. The need for this high level of integration argues strongly for a solution that includes such interfaces as part of its architectural design.

At the same time, an ideal licensing solution must be both modular and extendable in order to provide the flexibility needed to satisfy wide-ranging and rapidly changing market demands. Since no single provider of pricing, protection or licensing technology is likely to offer all required capabilities, the desired solution must be able to incorporate functional modules from a variety of sources. It must also allow publishers to assemble combinations of such modules at will, to fit their own unique and evolving needs.

These requirements clearly indicate the need to move beyond traditional, stand-alone electronic licensing technologies and establish a truly integrated platform for software licensing. Such a platform lets publishers save money by eliminating the inefficiencies of incompatible or poorly integrated components. By the same token, a modular platform makes it easy to combine capabilities in innovative ways. This in turn creates new revenue opportunities, by letting publishers price and deliver products to their customers in ways that would previously have been difficult or impossible to implement.

FLEXIBLE PLATFORM SOLUTION



The illustration shows a high-level architecture for an integrated software licensing platform. Starting from the bottom and working up:

- Layers for transporting user licensing rights and software usage data underlie both the core licensing engine and the “partner” business applications that form part of the platform. This allows the data to be readily shared.
- The “Connector” layer represents the interfaces that ensure seamless connectivity and communication between those applications and the licensing mechanism.
- Within the licensing system itself, sets of modular functionality allow access as needed to security technologies, policies, and usage management and tracking.
- A management layer provides a simple and convenient way for publisher applications to access the capabilities of the platform, without having to be concerned with what goes on in lower layers.

Implications of an Integrated, Modular Platform

By definition, an integrated software licensing platform provides—and, more importantly, defines—a standard way for interrelated business systems to connect and communicate. Because each “partner” system has been qualified for platform compliance, problems with compatibility and interoperability are effectively eliminated: everything just works.

Moreover, this standardization of interfaces allows new pricing, packaging, management and security capabilities to be “plugged in” as needed, in almost limitless combinations (see Figure 3). Because there are no restrictions on the number of capability-providers who can participate, the potential feature set of an integrated platform far exceeds that of any single product or technology.

FLEXIBLE INTEGRATED PLATFORM FOR SOFTWARE LICENSING

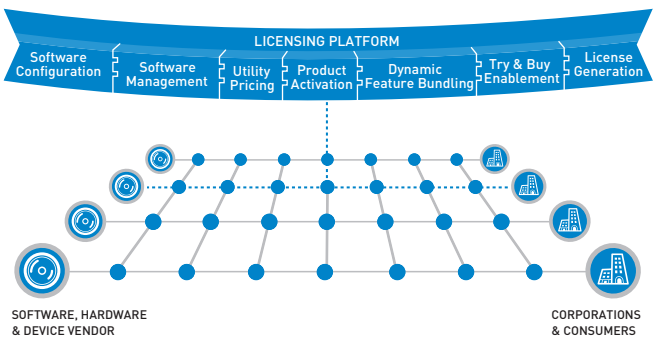


Figure 3: Modular and Extendable Platform

This combination of characteristics has immediate benefits for software publishers seeking to ease the effects of the “complexity crisis”:

- Standard interfaces and pre-qualified components make implementing and maintaining a licensing system much simpler, and therefore significantly less costly.
- The modular design allows for custom-tailored solutions to be designed and implemented quickly and easily. This lets publishers pursue more revenue opportunities more effectively in less time.
- The breadth of available capabilities provides publishers with “one-stop shopping” for software pricing, packaging, tracking, management and a wealth of associated functions.
- Modularity also allows a publisher’s different departments or business units to integrate specific parts of the platform with minimal consideration for the implementation choices made by other units—another cost-saving efficiency. For instance, the engineering department might implement the licensing API but leave productization choices to the product

manager. Similarly, the operations group can implement a licensing fulfillment solution that is independent of the choices made by engineering and development.

BENEFITS OF A SOFTWARE LICENSING PLATFORM

Revenue Enhancement

- Increased sales from enabling innovative license models like Utility Pricing
- Re-captured revenue from automatically enforcing compliance and eliminating unauthorized and illegal use
- Faster time-to-market for new offerings made possible by flexible, modular architecture

Cost Savings

- Lower cost and greater efficiency in implementing and maintaining license management, thanks to platform standards
- “One-stop shopping” for all needed capabilities
- Reduced operational and development costs and optimized upgrade paths resulting from modular application implementations

Increased Customer Satisfaction

- Fast, flexible response to individual pricing and licensing needs
- Reduced administrative burden for asset management and license compliance
- Optimized software expenditures because of reliable, accurate usage information

Investment Protection

- New licensing features and functions can easily be added whenever needed
- Systems will interoperate seamlessly now and in the future

Taking a longer-term view, the emergence of a standard licensing platform will enable software publishers to combine their own applications with those of other vendors into a single package. This will both broaden the appeal of the products involved and simplify the licensing process for customers. For publishers who embrace the platform, the opportunities for creating innovative partnerships will be truly revolutionary.

Modularity » Flexibility » Responsiveness » Revenue

Software publishers who implement and maintain licensing systems have more to contend with than just the growing complexity of their own infrastructures. Their customers, too, are faced with the challenges of diversification, globalization, and an uncertain economic environment. As a result, the demands of those customers regarding pricing, packaging, and licensing terms are steadily becoming both more sophisticated and more individualized.

A licensing platform that draws upon a wide range of functional modules from a variety of sources gives publishers maximum flexibility to sell products the way their customers want to buy them. Because each module is pre-qualified for conformance to platform standards, implementing new pricing and licensing models is fast and easy. This lets

publishers respond quickly to customer needs, even to the point of offering unique terms to individual customers. That, in turn, means happier customers—and more of them.

For example, one customer might be offered a site license, another could be issued “named user” licenses, and yet another provided with “floating” or “concurrent” licenses. A single customer could even be offered a variety of options in combination, without changing the underlying structure of the licensing system. The modular architecture and tightly integrated nature of a licensing platform enable a level of customer responsiveness that has never before been possible to achieve.

The Appeal of “Utility Pricing”

Perhaps the ultimate in pricing flexibility is the ability to license software on a “pay as you use it” basis. Software buyers favor this model because it aligns value with cost. In addition, when companies know they can pay for software according to actual usage, they have little reason to restrict availability to a small number of “seats,” and are happy to extend access across their entire organization. That, in turn, explains why publishers like this model too—greatly expanded penetration of their products within accounts. Industry analysts report that interest in “utility pricing” is sharply on the rise among software purchasers, indicating that the “pay only for what you use” pitch has huge potential for attracting new customers and retaining current ones.

A key challenge of pay-as-you-go pricing is that usage data may need to be shared among a variety of business systems in order to provide timely and accurate tracking, management, fulfillment and billing. This once again raises the specter of costly integration and implementation issues. The answer: a licensing platform whose components have already been qualified to work together seamlessly, thus eliminating those concerns. The “platform” approach makes the implementation of highly attractive utility computing models not only achievable, but simple.

Management and Compliance: Publishers Get Paid When Customers Have it Easy

Businesses are feeling the pain of determining how many software licenses they should buy for each software product they use, and then tracking the use of those licenses across multiple sites and geographies. The problem has become so severe that some corporations refuse to buy software from publishers who don’t make it easy for them to address their own asset management “complexity crisis.” For example, Ford and Motorola insist that the software products they buy provide FLEXIm[®] licensing. For the publishers themselves, it’s an even more fundamental matter of dollars and cents: having customers who can’t or don’t comply with licensing terms means revenue is going down the drain.

Although companies generally want to stay compliant, it is increasingly hard for them to do so. One of the main reasons is the complexity of licensing terms, which differ greatly among publishers and are often ambiguous or confusing. Another major hurdle is the lack of staff available to perform required audits (or the company's unwillingness to allocate staff resources because of cost).

“With software license compliance being a major concern for all companies, users are looking for ways to stay compliant and have flexible pricing that reflects their actual usage. While hardware vendors have moved toward utility pricing models, there are still significant opportunities for software publishers to optimize their revenue opportunities and avoid the cost of expensive audits,” says Patricia Adams, Senior Analyst at Gartner, September 2003

If publishers expect to be fairly compensated for their products in this difficult environment, they must make compliance easier for their customers. From the customers' point of view, they need:

- A way to monitor their usage against licensing terms and maintain compliance automatically.
- Generation of data that lets them manage their assets centrally, and realize cost savings by optimizing their usage.
- A means of making these tasks completely transparent—“hidden under the covers”—so that they never have to worry about them.

Here again, an integrated platform offers striking advantages over isolated technology in meeting these requirements. By ensuring that all components that send, receive or track usage data are qualified and platform-compatible, implementation cost is reduced, reliability is increased, and transparency is maximized. Using such a platform to implement an effective asset management and compliance monitoring system creates a classic win-win situation. Customers are happy because they can easily determine what they have and when they need to buy more without having to decipher cryptic licensing terms or endure costly audits. Publishers are happy because their customers are happy—and because they recover every dollar of revenue they are entitled to!

Source-Level Modularity as a Cost-Reduction Strategy

An ideal license management platform allows software publishers to extend the concept of modularity into their own code. In doing so, publishers can realize significant cost savings while in many cases increasing customer satisfaction.

For example, a publisher could create “lite,” “standard” and “premium” versions of a product, all contained in a single software binary. The publisher could then distribute all these product versions as a single, electronically controlled package, yielding a significant reduction in operational costs. Customers could purchase upgrades instantly using a simple web-based interface to the licensing system, saving them time and money as well. This kind of convenience would certainly make the product more attractive to purchasers.

A more ambitious publisher might use licensing platform tools to make a large number of product features individually licensable. While this may require more effort, the pay-off is correspondingly great. Licensing individual features gives publishers the ability to recombine feature sets and quickly meet changing market requirements without changing their code, leaving engineering teams free to focus on new development. That's clearly better than expending valuable resources to turn out a new product release just because the marketing department wants to offer a different feature set.

The software licensing category is cluttered with small vendors claiming licensing solutions. Macrovision® has held the dominant market position for years with its FLEXIm® licensing application and many believe this to be the de facto standard in the software industry. Recently the company announced the industry's first licensing platform, Macrovision FLEXnet™. In addition to offering a modular system for pricing, packaging and protecting software applications, the company has established a broad network of developers and suppliers that offer products that interoperate with their licensing platform.

The Value of Being “Future-Proof”

Given that the future is unpredictable, software publishers must do what they can to make sure the technology they buy today will continue to add value tomorrow. Here, most clearly of all, it makes sense to invest in a licensing platform as opposed to individual tools or applications. By providing a modular, extendable architecture encompassing a wide variety of capabilities from multiple sources, a flexible platform ensures that new features and functions can be added easily whenever needed. And by requiring that modules be qualified as compliant, the platform approach gives publishers confidence that their systems will continue to interoperate smoothly as they evolve, thus avoiding costly surprises.



Gartner Research Note
 April 17, 2003, Jane Disbrow

Clearly Define Software License Usage Rights

Software license usage rights often are overlooked during software license negotiations. During audits, enterprises may discover that software vendors interpret “usage” differently than anticipated.

During a software license negotiation, software usage rights often are not given the scrutiny that they should receive. An enterprise may be unaware that its software vendor has a different interpretation of usage rights until a software compliance audit. This is not a favorable time to start such discussions. No enterprise wants to sue a software vendor because of terminology issues; software vendors also do not want to take their customers to court. Although most software license agreements are written by the software vendor, with the vendor assuming the primary responsibility for defining such usage rights, both parties benefit by ensuring that usage rights are clearly defined in software license agreements.

In December 2002, Gartner surveyed customer relationship management (CRM) application vendors about license models. The results shown in Table 1 derive from 89 responses.

CRM VENDOR LICENSE MODELS

LICENSE MODEL USED	PERCENTAGE	COUNT
Named User	42.7	38
Named User per Module	20.2	18
Per Module	65.2	58
Concurrent User	27.0	24
Revenue or Other Financial Metric	9.0	8
Number of Employees	3.4	3
Transaction-Based Pricing	7.9	7
Role-Based Pricing	12.4	11
Size of the Hardware Platform	13.5	12
Other	36.0	32
Multiple License Models per Software Vendor	237.3%	211

Source: Gartner Research [December 2002]

Table 1: CRM Vendor License Models

Named User License Model

This is a common license model and can be one of the easier to understand metrics if usage rights are clearly defined. Many enterprises believe a “named user” is a person who has usage rights to the software. In most cases, this is a valid assumption. However, this can become complicated if:

- The software vendor includes a clause stating named users also include nonhuman devices, but examples of nonhuman devices or how they are to be counted are not defined.
- The contract states that usage must be counted as the “multiplexing” front end, but the contract does not clearly define when or how that usage would be counted.
- The contract states that named users who “directly or indirectly” use the software must be counted. Any “indirect” access to the software can be broadly or narrowly defined (similar to the multiplexing front-end issue).

These definitions open to interpretation what must be counted as a named user. “Multiplexing front end” or “indirect use” may not be stated in the contract, but may be based on how the software is used. The vendor may audit and then state that the “actual” usage of the software is more extensive due to access via another software front end. If the software vendor has stated that “nonhuman devices” must be counted, the enterprise should require that the vendor perform an analysis of the current environment and include examples of nonhuman devices that should be counted in that environment. If the right to generic logons is necessary, ensure this is contractually defined and allowed in the software license agreement.

Clearly address in the agreement how usage will be counted in case of an audit. If accurate usage cannot be measured, changes should be made to the usage rights. Definitions of multiplexing front ends, nonhuman devices and indirect use can vary among software vendors. The contract’s definition must be sensible and measurable.

Concurrent User License Model

This previously was a common license model, but fewer software vendors are offering it. One reason for its decline is that it was ill-defined in prior contracts, which caused problems for vendors and their customers because both parties believed that their interpretation of a “concurrent user” was correct. Examples where usage rights may not be clear in a contract include:

- Will a user count as one concurrent user or multiple concurrent users if he or she is logged onto several screens in a system, while most likely only “active” in one screen?
- Will a user count as a concurrent user if he or she is logged onto the system, but is not actively using it?
- How is usage counted if the software vendor has usage rights that state that measurement must be counted at the multiplexing front end (similar to the named user model)? This question also applies to nonhuman devices and indirect usage.

Several enterprises use internally developed software that takes “snapshots” of usage. However, this may underestimate usage, or at least usage as the software vendor defines it. The software license agreement may state that usage must be counted at a multiplexing front end, but how that usage will be counted or measured in case of an audit may not clearly defined. If some generic logons will be used, the rights to have such logons must be allowed in the contract usage rights.

Server-Based License Model

The definitions of “server” and “computer” often are not clearly defined in the contract, and issues such as several instances of the software running on one server or computer are not addressed.

For example:

- Does the enterprise need to have additional licenses if it runs more than one instance of software on the same server?
- Are there usage rights restrictions based on the size of the server, or limited to only one platform or equipment serial number?
- Does the “server” license include rights to run the system on a mainframe?
- Does the enterprise have the right to run another copy of the software for backup or test purposes?

Gartner reviewed a recent license agreement that stated that the customer was licensing a quantity of three server licenses, with a restriction of “up to four CPUs per server.” It was not clear if the server had to be restricted to only running four CPUs or less, or if the software could be

installed on any size server as long as the software was not installed on more than four CPUs. These issues should be covered in the license agreement.

Role-Based License Model

Several software vendors offer role-based license models, such as “professional,” “employee,” “casual,” “normal” and “super” users. If there are two license definitions, usage rights generally are more easily understood (except for issues similar to what occur with named user licenses). However, where there are three or more user categories, the “middle” category often is not clearly defined. In some cases, there are special considerations for that middle category, in which an oral agreement provides that certain departments or user categories can use it. However, this often is not articulated in the software license agreement.

It also can be difficult to maintain the number of licenses required for each category. This is compounded by the number of user categories. It will be easier to comply with usage rights if the contract includes provisions to convert from one license model to another at some point in time. For example, an enterprise that licenses 1,000 professional, 500 limited-professional and 500 employee self-service user licenses should be able to swap, value for value, license types each year.

Employee-Based License Model

An employee is an employee, right? Not exactly. Before agreeing to an employee-based software license agreement, enterprises should discuss the definition of an “employee” with their human resources organizations to clarify issues, including:

- If the enterprise has many seasonal workers, will license fees be based on the maximum number of employees during a year or on the average number of employees, or the number of employees at one point in time, to represent a base number of employees?
- If the enterprise has high employee turnover, will the number of employees be based on the total number of income tax documents issued or the average number of employees?
- If the enterprise has many part-time employees, will the software vendor be willing to use a full-time equivalent formula (that is, 40 hours worked equals one full-time equivalent)?
- How will temporary workers, employees on leaves of absence, retirees and nonactive employees be counted?

Financial-Based License Model

The license model encompasses models based on total revenue, budgets or cost of goods sold, for example. Some software vendors use these metrics to price software, even if this use of the metrics is not clearly stated. Financial-based contracts often include these inadequate definitions:

- “Revenue” shall mean the gross revenue reflected in an audited statement from its external accounting firm.
- “Budget” shall mean the gross budget reflected in an audited statement from its external accounting firm.
- “Cost of goods sold” is defined as the total cost of inventory that an enterprise sold during its fiscal year.
- “Transportation spend” is defined as the total cost for transportation services during an applicable year.

Revenue-based licenses models face problems, including:

- One-time occurrences that do not reflect the average budget or revenue during some period of time. If the software vendor has built in “upgrades” based on increases in financial-based systems, an upgrade would have to be paid even if the metric dropped the following year. Generally, the software vendor does not build in provisions to handle one-time events or reductions in financial-based systems.
- There may be “pass through” costs that do not reflect true revenue or budget – for example, a utility that buys electricity when needed and resells it immediately to a subsidiary or other entity.
- Charges as “fuel tax surcharges” may increase for a period of time, but there is no additional value received from the software. If gasoline costs go up significantly in the next year, then the “transportation spend” cost also will increase. However, the basic value of the software to the enterprise has not changed.
- Financial measurements such as “budgets” may reflect different allocations of expenses each year; however, the basic budget stays the same.

Before agreeing to a financial-based license model, enterprises should work with the department that “owns” that license metric to ensure that there are clear definitions of what will be included in the metric and what should be excluded. Build provisions into the software license to handle downturns in any financial-based agreement.

Transaction-Based Pricing License Model

Transaction-based pricing makes sense for some licensing metrics; however, the metrics must be clearly defined, as well as what will be counted. For example, before an enterprise agrees to a metric based on “number of purchase orders” or “number of purchase-order lines,” it must work with its procurement department to examine samples of

purchase orders to determine if this metric must exclude certain criteria—for example, a zero-dollar purchase line or purchase order. If the enterprise has a large number of small purchase orders, this often can be used as leverage to get a higher discount to purchase such licenses from the software vendor.

Bottom Line

There is no perfect license model. All models have advantages and disadvantages. Any license model must ensure that usage rights are clearly documented in the license agreement, as well as how usage will be measured. The model must consider what will happen if an enterprise needs more licenses (for example, price protection for a period of time and functionality replacement), as well as fewer (for example, right to remove licenses from maintenance and support with a corresponding reduction in ongoing maintenance and support). Gartner believes that it is unreasonable to license for anything that cannot be accurately measured. An enterprises should not wait until an audit is in progress to discuss usage rights with its software vendor. Understand how the software vendor priced its software, as well as all the metrics involved in the pricing. Clearly define those metrics and usage rights, and understand how they will be counted in case of an audit.

©Macrovision Corporation 2003.

Macrovision and FLEXIm are registered trademarks and Macrovision FLEXnet are trademarks of Macrovision Corporation.

Overcoming the Software Licensing Complexity Crisis is published by Macrovision Corporation. Editorial supplied by Macrovision is independent of Gartner analysis. All Gartner research is ©2003 by Gartner, Inc. and/or its Affiliates. All rights reserved. All Gartner materials are used with Gartner's permission and in no way does the use or publication of Gartner research indicate Gartner's endorsement of Macrovision products and/or strategies. Reproduction of this publication in any form without prior written permission is forbidden. The information contained herein has been obtained from sources believed to be reliable. Gartner disclaims all warranties as to the accuracy, completeness or adequacy of such information. Gartner shall have no liability for errors, omissions or inadequacies in the information contained herein or for interpretations thereof. The reader assumes sole responsibility for the selection of these materials to achieve its intended results. The opinions expressed herein are subject to change without notice.



macrovision™

Macrovision Corporation
2830 De La Cruz Boulevard, Santa Clara, CA 95050
US +1 888-755-0861 Int'l +44 (0)870-873-6300
www.macrovision.com

STGWP.0104