

# The Evolution of Software Licensing Models

## Introduction

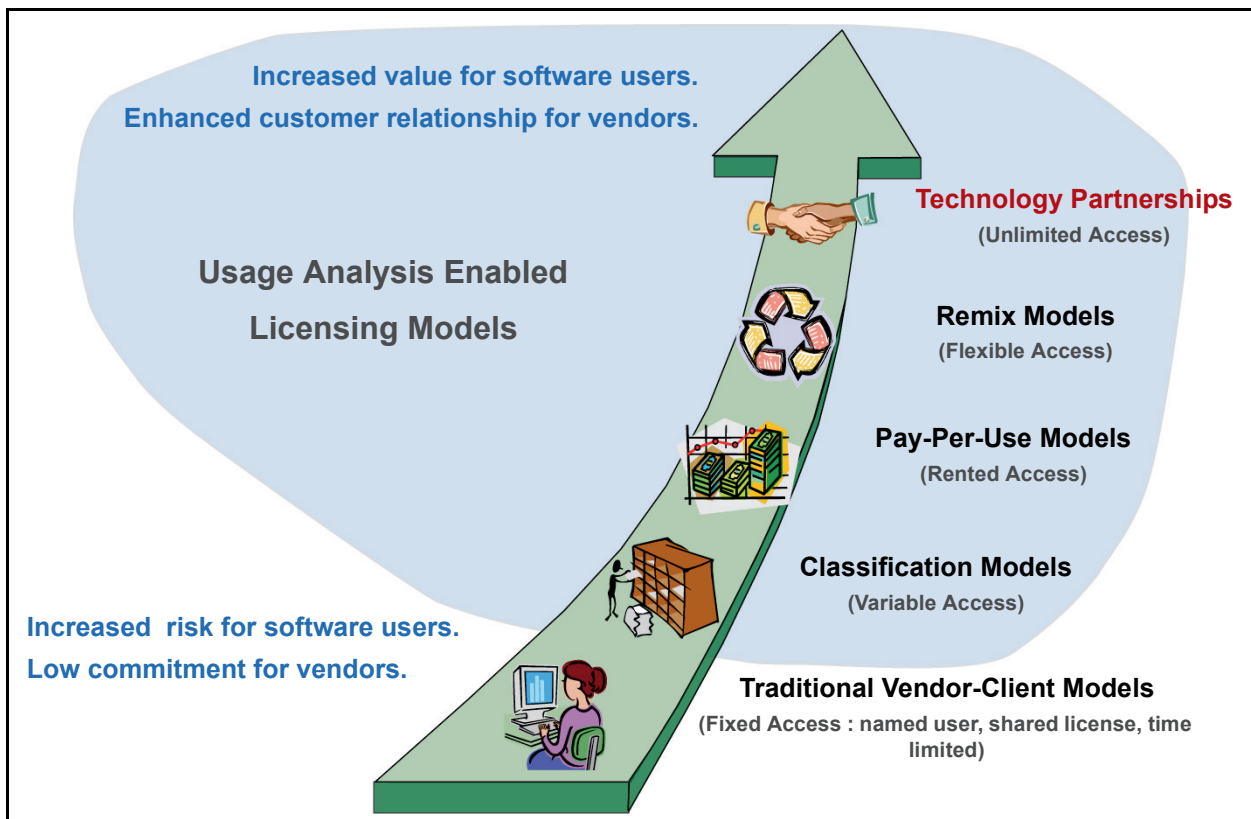
Software continues to play an increasingly critical role in business. In an effort to enhance customer satisfaction and value, software vendors have adapted to changes in surrounding technology and are offering a wider array of licensing options to their customers.

Traditional vendor-client models, including single user-single license, multiple users-shared license, and temporary or fixed period licenses, are well understood and widely used. Although traditional models have evolved with technology innovations, they do not fully satisfy the business issues faced by today's enterprises; issues such as balancing productivity and efficiency, estimating software needs, adjusting to changing needs, and

dealing with new requirements late in a fiscal year.

Detailed analysis of actual software usage, using tools such as **License Tracker**, has proven to be an enabling technology for many new licensing models which do address these issues. As shown in the diagram below, these models range from user classification through pay-per-use and product family remixing to technology partnerships, and provide increasing value to technology consumers with a corresponding increase in commitment to the software vendors.

This paper provides an overview of the various licensing options, their respective benefits, and the importance of understanding usage.



## Traditional Vendor-Client Models

### Single User Licenses

In the traditional vendor-client model, the software consumer purchases a license for each user that needs access to the software. These licenses may be either assigned to a named user or node-locked to a particular computer.

The single node perpetual license is conceptually quite simple from both the vendor and client perspective, and is easy to manage.

For the manager making software purchasing decisions the process is easy for low cost software; buy one copy for each employee that needs to use the software. The process becomes far more complicated for expensive software where fewer licenses and system sharing (possibly through shift work) become considerations to maximize value for software dollars. The situation becomes even more complicated with larger companies where the business needs of their various groups change from time to time.

### Multi-User Shared Licenses

In the late 1980s, the introduction of networks gave rise to new licensing technologies allowing companies to share licenses between employees.

This model gave managers a means to share expensive software without having to share physical computers. Although multi-user licenses are typically more expensive than single user licenses, the enhanced value to the consumer justifies the difference.

Concurrent user licensing is fundamental to most of the new usage analysis enabled models. By itself however, it does not address the reality of ever changing business needs within the enterprise.

### Time Limited Licenses

The "demo license" has become a standard tool for allowing potential customers to evaluate software prior to making a purchase.

The extension of this technique to provide fully functioning software for a set period of time (normally one, three or twelve months) provides a mechanism for managers to deal with short-term and variable requirements. These short-term licenses work very well when:

- the client needs extra capacity in specific times of the year
- the client is unsure of the need for the software and wants to test internal demand
- the client wants to expense their software usage, deferring the expenditure over time, rather than capitalize the purchase

Again, the consumer has seen an increase in the relative cost of software (most often the cost of only a few short-term licenses is equivalent to a single perpetual license) but they have also seen a corresponding increase in value to the expenditure of their software dollars.

These licenses have also been used by some vendors for peak demand satisfaction with 1 week license periods.

*"In order to maximize the value of your software expenditures you need to have the right licensing arrangement with your vendors, including having the correct number and optimum type of licenses.*

*A proper understanding of how you are currently using your software is a prerequisite to having such discussions."*

## Motivation for New Models For Software Users

Software managers are tasked with balancing the conflicting goals of productivity and efficiency. Denial of access to core software can have a negative impact on productivity; however, purchasing an expensive software license that may only be used 1 or 2% of the year is highly inefficient. Access to a short term temporary license can deal with the inefficiency, but productivity can still be impacted by the delay between detection of need and availability of the temporary license.

Predicting future usage requirements at the time of making a purchase can be difficult, especially when dealing with a new vendor offering unfamiliar products. Flexible access during evaluation and assessment periods allows high software availability for unknown loads without pre-purchasing unneeded copies.

At the start of the fiscal year capital expenditure budgets tend to be firmly defined. New licensing models facilitate the mid-year acquisition of new software through operating budgets.

## For Software Vendors

The issues discussed above represent opportunities for vendors.

Competition between software vendors requires continual efforts to enhance customer relationships while seeking to increase market share by establishing oneself as the customers' first choice.

Early adoption of innovative licensing approaches can be an attractive differentiator. Late adoption of new licensing models can result in a reputation as a laggard and diminished customer relationships.

## Software Usage Analysis Enabling the New Models

The advent of multi-user shared licenses brought with it the notion of usage logging. Most concurrent use license managers provide usage logs with varying degrees of information; these logs almost always include details of who used what software, for how long, as well as details of any denials of access.

The existence of this usage data has led to the development of a new breed of software tools to analyze the usage data. Many of these tools are custom in-house developed scripts while others are fully supported commercial products like **License Tracker**.

Analysis of the usage data provides software customers with the ability to perform capacity planning, optimization of existing software license counts and usage, and departmental chargebacks, cost splitting and budgeting.

The analysis also provides software vendors with the ability to support new licensing models, and facilitates their customers' monitoring of the costs associated with these new models.

## Classification Models The Essentials

User classification is an extension of the well understood named user licensing model.

In this model, users are placed into categories defined by usage rights and restrictions including:

- which features can be (and/or can't be) used
- the duration for which certain features may be used
- how many sessions of particular features can be active in any given period (day/week/month)

Software users purchase named user licenses for the various categories defined by the software vendor. Frequently vendors will define custom categories for specific customers during contract negotiations.

Analysis of usage data is needed to monitor the actual category that all users fall into, and to determine if the enterprise is compliant with its agreement or if a "true-up" is required.

## Pay-Per-Use Rental Models The Essentials

The pay-per-use model entails the software vendor providing the customer with more licenses than those which the customer has purchased.

Use of all software, both owned and rented, is recorded in the usage logfile. Once per billing

cycle the logfile is sent to the vendor for analysis.

Post analysis of the license usage patterns allows the vendor to determine how much use of these extra licenses was actually made, and a pay-per-use invoice can be generated.

## Time Based Pay-Per-Use

In a time based pay-per-use arrangement, technology consumers are charged for the amount of time that they used non-owned copies of the software.

Immediate availability of the rented software provides on demand access such that end users never know if they have owned or rented copies.

### Pay-per-use Example : Oil and Gas

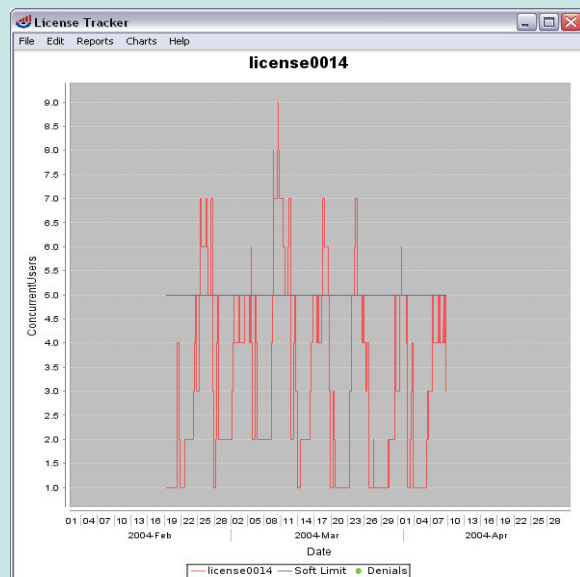
A major oil and gas company (OilCo) uses various products from a software vendor (SoftCo). The seismic analysis group at OilCo has experienced many occasions where attempts to start ProductX from SoftCo are denied by the license management system as all existing licenses are in use.

OilCo has the following options:

- purchase additional licenses of ProductX
- purchase short-term licenses for ProductX in case this is a temporary phenomenon
- enter into pay-per-use licensing arrangement with SoftCo

Merely knowing that there are instances of application startup being denied does not provide OilCo with enough information to make a proper long-term decision. So, OilCo enters into a short-term pay-per-use arrangement with SoftCo and purchases **License Tracker** to perform an analysis of the log files generated by the use of ProductX.

The analysis shows that OilCo consistently uses 2 licenses more than they own and occasionally 4 or 5 more.



Accordingly, OilCo enters into a long-term agreement with SoftCo, purchasing 2 more licenses and having 3 more available on a pay-per-use basis.

Ongoing analysis by OilCo using License Tracker will help them determine if and when it is appropriate for them to increase their paid versus rented license count.

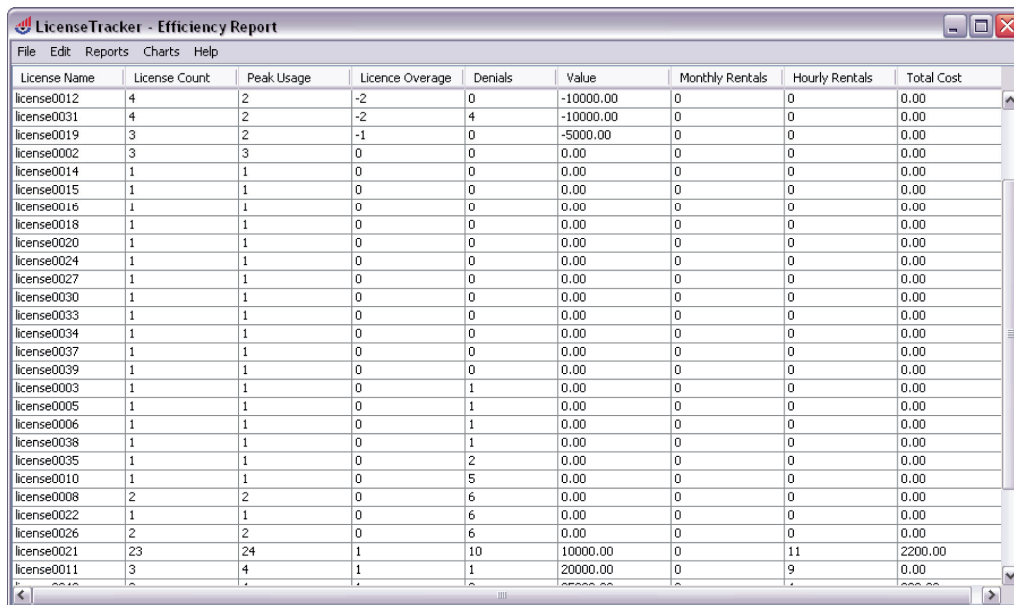
## License Tracker Support for the Remix Model

The License Efficiency report provides details of:

- number of licenses owned
- peak actual concurrent licenses used
- number of unused licenses (owned - peak))
- number of denials
- number of rented licenses
- total rental costs

This information is provided for each licensed software application that is tracked.

A license analyst, after reviewing this report, can make informed decisions about which licenses to remove from the mix, and which licenses to add to the mix, for the upcoming period.



License Name	License Count	Peak Usage	Licence Overage	Denials	Value	Monthly Rentals	Hourly Rentals	Total Cost
license0012	4	2	-2	0	-10000.00	0	0	0.00
license0031	4	2	-2	4	-10000.00	0	0	0.00
license0019	3	2	-1	0	-5000.00	0	0	0.00
license0002	3	3	0	0	0.00	0	0	0.00
license0014	1	1	0	0	0.00	0	0	0.00
license0015	1	1	0	0	0.00	0	0	0.00
license0016	1	1	0	0	0.00	0	0	0.00
license0018	1	1	0	0	0.00	0	0	0.00
license0020	1	1	0	0	0.00	0	0	0.00
license0024	1	1	0	0	0.00	0	0	0.00
license0027	1	1	0	0	0.00	0	0	0.00
license0030	1	1	0	0	0.00	0	0	0.00
license0033	1	1	0	0	0.00	0	0	0.00
license0034	1	1	0	0	0.00	0	0	0.00
license0037	1	1	0	0	0.00	0	0	0.00
license0039	1	1	0	0	0.00	0	0	0.00
license0003	1	1	0	1	0.00	0	0	0.00
license0005	1	1	0	1	0.00	0	0	0.00
license0006	1	1	0	1	0.00	0	0	0.00
license0038	1	1	0	1	0.00	0	0	0.00
license0035	1	1	0	2	0.00	0	0	0.00
license0010	1	1	0	5	0.00	0	0	0.00
license0008	2	2	0	6	0.00	0	0	0.00
license0022	1	1	0	6	0.00	0	0	0.00
license0026	2	2	0	6	0.00	0	0	0.00
license0021	23	24	1	10	10000.00	0	11	2200.00
license0011	3	4	1	1	20000.00	0	9	0.00

A detailed discussion of the contractual and operational considerations of pay-per-use can be found in the whitepaper, "*The Mechanics of Pay-Per-Use Licensing*".

## Transaction Based Pay-Per-Use

In a transaction based pay-per-use arrangement, usage charges occur because a software module has been used. The duration of use is irrelevant.

This model is most applicable to situations where individual user functions take a consistent amount of time. The number of times each module or function is used will determine the invoice amount.

## Remix License Models The Essentials

The problem with all previous models is that they do not consider the changing needs of business. As the market conditions change, so do the needs of business and the technology they want to access.

The remix model brings a tighter business relationship between the vendor and the consumer. In this model, the software consumer makes an investment in the vendor's technology in general, not in specific licenses. The consumer is permitted to access a wide range of software products from the vendor with the ability to adjust the number of licenses (i.e. a **remix**) to balance the software



accessed to the needs of the company within the aggregate value of the technology pool.

Remix works well when the customer is looking for **long-term value** from the vendor rather than lowest immediate cost.

## Static Remixing

Static remixing can be done without modification to either the software applications or the license manager.

In such an arrangement, the technology consumer assesses their usage patterns and anticipated upcoming needs and selects a new license mix periodically (every 3 or 6 months).

Usage analysis provides the historical information needed to make informed remix decisions.

## Continuous Real-time Remixing

This model is also known as token based licensing. Instead of purchasing licenses for individual features in a vendor's product family, the consumer purchases licenses for a generic token feature. Each of the actual software features are assigned a relative value in terms of tokens.

When the actual features are checked out, the corresponding number of tokens are checked out.

Pay-per-use can readily be added to this model by providing access to non-owned licenses of the token feature and processing the logfile as described earlier.

*"License Tracker has been invaluable in managing our exploration licenses.*

*By understanding how we are using our concurrent licenses we can more effectively budget, negotiate with vendors, and provide our users with optimal licensing."*

B. Peers  
Team Leader, Applications Support  
Exploration Technology  
Talisman Energy Inc.

In order to move an existing product family to this model requires either:

- modification of the applications to add checkouts and checkins of the token feature, or
- use of a license manager with support for token licensing

## Technology Partnerships The Essentials

Technology partnership arrangements provide the consumer with unlimited access to the vendor's technology; they are sometimes referred to as "buffet" or "all-you-can-eat" deals.

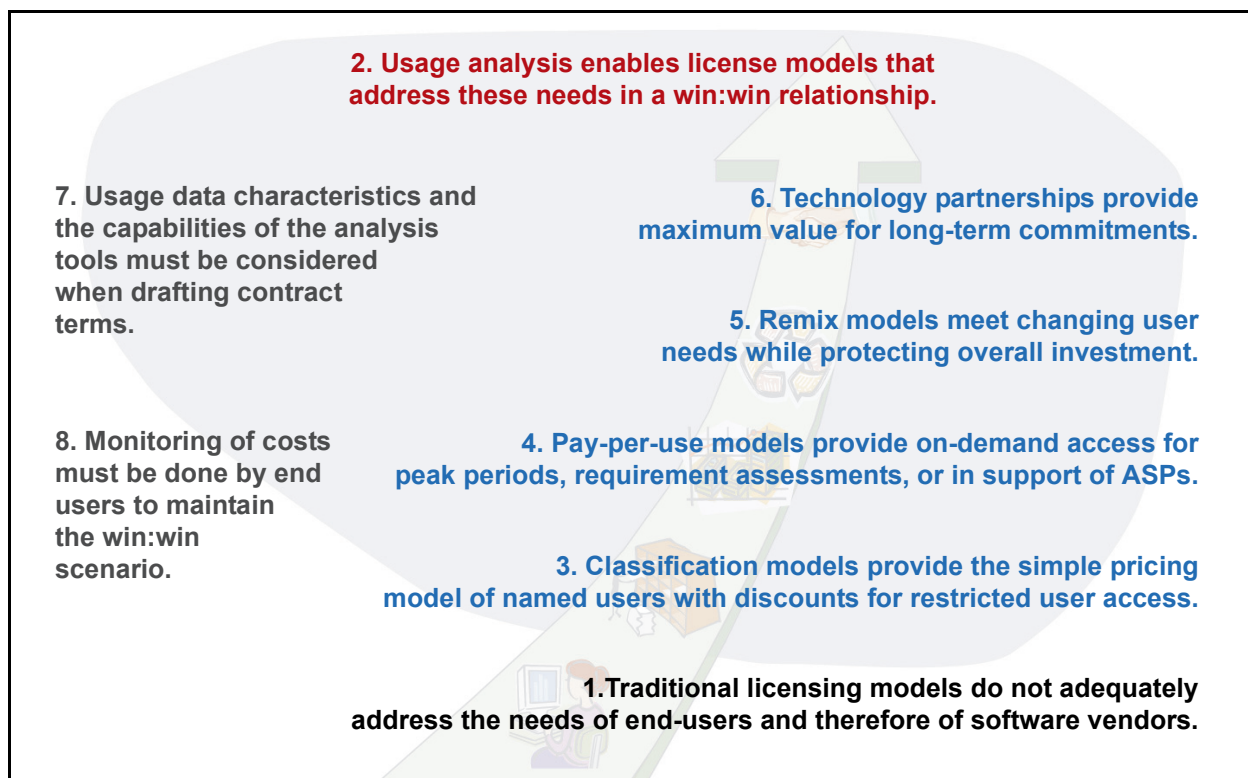
This model can be most rewarding as it aligns the interests of both the vendor and the consumer. As the business needs of the consumer change, the vendor provides the technology to meet those needs. The client derives significantly more value from the vendor, while the vendor gets both a level of guaranteed revenues as well as improved technology adoption within the client resulting in improved revenues.

Agreements for this model are multi-year contracts where the consumer pays a fixed annual fee for unlimited access to the vendor's technology (or a defined portion thereof).

The annual fee for each year is adjusted based on a pay-per-use analysis of actual usage in the previous year and an agreed formula (i.e. minimum amount fee for the upcoming year will be based on average usage for the last x quarters of the previous year).

## Other Considerations End User Cost Monitoring

Whereas vendors only require access to usage data once per billing cycle for invoicing, end-users must continually monitor usage patterns and their corresponding costs.



Regular monitoring of usage and costs can allow consumers to:

- ensure actual costs are not exceeding budgets
- detect and correct improper license use
- make mid-period purchases if rental usage is higher than expected

- access : how and when is the logfile sent to the vendor by the customer
- privacy : privacy restrictions may require usage logs to be scrubbed before being sent to vendors

## Maximizing the Value of Your Software Expenditures

### Usage Data Issues

The usage data to support these newer licensing models is typically generated by the license manager. It is also possible for usage data to be generated directly by the application software.

Contracts for new licensing models must consider these usage data issues:

- existence : generating logfiles is sometimes an option that must be turned on; improper file management can result in logfiles being overwritten or deleted

Success in business requires that we maximize value for the dollars we spend. At some times this can mean minimizing immediate costs, while at others it means increasing long-term effectiveness.

The needs for each company are unique. The requirement to understand how your software assets are being used is necessary for proper decision making.

New tools (like **License Tracker**) are enabling software companies to provide a wide range of licensing options to their customers, thereby helping them to maximize the value received for their software investments.

---

*"You can only optimize -  
what you understand..."*

*...and you can only understand -  
what you track!"*

---



[www.licensetracker.ca](http://www.licensetracker.ca)