



Buy cheap, buy twice – don't waste your money when purchasing software

By Judica Krikke

We have just closed a fantastic deal. Or was it a huge waste of money? Remarkably, this is often hard to determine when purchasing software. We seem to have obtained a substantial discount. But is this really the right product for the intended purpose? Will this purchase result in the hoped-for acceleration of business processes, improved service, or cost savings? Have we bought too many licences and paid too much? Do we have enough licences? Will there be any unforeseen costs?

In the world of software a bargain can often turn out to be a costly mistake. A discount can seem attractive at the outset but often masks issues which eventually lead to higher costs than expected. For example, the customer is assessed on licensing rules that he did not understand at the time of purchase, or which he just did not pay attention to because his eye was on securing a cheap deal. There can be hidden issues such as a requirement for additional software, perhaps the software needs to be modified due to technological changes, the licensing model has to be adjusted, or the product is no longer supported at the expense of the buyer.

'Hinges, doors are not included'

Sometimes there are kitchen salesman-like scenes: 'fantastic discount only for you if you sign today' actions. As with the purchase of a kitchen, it is recommended that you question and read the small print before ticking the box. After all, it will be fairly disappointing if you later discover that hinges, doors, the oven and the refrigerator were not included in the deal. Or when you find out that the kitchen's earliest delivery is about three months, but you are moving in your brand new house within a month.

Of course, this is stating the obvious. Any buyer ought to understand what he is buying before purchasing and not try to figure it out after the event. But, staggeringly, for some reason this does not seem to happen when buying software. Are purchasers simply not being careful enough or is there more to it? Although sometimes major discounts undoubtedly cloud the purchaser's decision, the answer is that there is more to it.



Purchasing software is a complex matter in a complex landscape. The first complexity layer lies in the nature of the product. The mapping of needs and determining which product best meets those needs is not always easy. This identification takes time and requires expertise, both from the software supplier and the (potential) customer. Knowledge of the organization of the customer must be combined with knowledge of the products and the user experience. It is the supplier's responsibility to inform the customer properly and ask the right questions to gain a thorough understanding. The potential buyer has a similar responsibility to be clear about what he needs.

Once it is determined which product is best, then the appropriate licensing or service model should be selected. Here we find the second layer of complexity. Which licence model is required and how many licences of any type should be purchased? For example, is it more beneficial to purchase licences based on the number of users, or is it better to use the hardware environment as a standard for an accounting model? For how many users or devices should a licence be purchased? Is it better to choose a Software-as-a-Service model or a traditional licence? What is the corresponding payment model and how does that work out for the buyer?

That may not sound particularly complicated. Often licences are simply based on counting the number of users and hardware devices on which the software is installed. Unfortunately, the reality is different. Some software suppliers handle extremely complex models, which are far from transparent. Therefore, the licence need is sometimes difficult to determine, especially for a customer who is not familiar with the various models available.

'Lack of transparency and diversity in licensing and user models'

It is even difficult for suppliers of the software or the retailer who sells the software on behalf of the supplier to foresee all issues. For example, a licensing model that is based on the number of users is highly dependent on how the term "user" is defined. Is it the number of users or the total number of concurrent users? Is the number of users determined by number of FTEs or individuals? Can anyone use it or only the persons mentioned by first and last name? Is a licence required for all persons who actually work with the software? Or for all individuals who have the ability to use the software? What about access for contractors: may they use the software? Is that too complicated and is a licensing model based on the underlying hardware easier to understand? On how many servers is the software installed? What is the licence-measure: server, CPU, processor core? How should they be counted? How will it be counted in a virtual environment? Are all the laptops, tablets



and smartphones included in this process? Please note these questions concern simple licensing models and the list is far from exhaustive. It can be a lot more complicated because of different user groups and mixed licensing models. On the other hand, it can be a lot easier with, for example, a fixed fee or an annual fee based on the turnover of the customer. Still you must pay attention because on which turnover do we actually focus?

Failure is unavoidable if the buyer does not exactly understand the licensing model that he purchased beforehand and if, subsequently, the application of the software is not correctly set up. If too few licences have been purchased it may not be evident until the first software audit, and if too many licenses have been purchased it could result in overpayment. Moreover, the annual cost of maintenance - which is often based on the initial purchase – may end up being much higher than necessary.

As an owner of the software, the supplier is entitled to payment for the use of its products. That is his established right. However, as a result of the lack of transparency and complexity of licensing models for the customer it is not always clear when he has used more than is permitted. The supplier relies on non-compliance with the conditions and on violation of his intellectual property rights of the software. The software supplier can be compensated for use of the software beyond the limits of the licence on both grounds, often leading to hefty fines and removal of the favourable discount offered at the initial stage of the transaction.

In addition, the supplier may threaten to suspend support and/or withdraw licences altogether. These repercussions are usually included in the small print, which is often blindly accepted by the customer. This takes us to the third layer of complexity. Business operations are often highly dependent on the software. The supplier has several means to put pressure on the customer when he has not complied with the licensing rules - even though he might not have fully understood them nor had any intention of exceeding the limits. Out of fear, new licences are often bought quickly with little understanding of the exact purchase. At the end of the day, it is a question of supply and demand. The supplier and its distributors are under a legal obligation to inform the potential customer carefully before any purchase is made and they may be held accountable for doing so. They should advise on the licensing models and provide an adequate explanation. Is the explanation incomprehensible? Do not accept the situation if a supplier is unable to explain precisely how his licensing model works as this means something is wrong with the supplier and not with you.

Don't accept mumbo jumbo and do not accept being put under pressure to purchase immediately because the discount will expire otherwise – you don't fall for that when the kitchen seller applies the same technique. Do not be fooled into signing there and then, take time to read the conditions, let them explain the small print, make sure you understand what you are getting involved with. If this means less discount then so be it, but it should also mean that there are no excessive bills afterwards. Use all available resources to force the supplier to be transparent and insightful regarding the actual costs ahead. Is it not clear? Don't just hope that it will be fine, because it won't: in a software audit you will not meet a kind account manager, instead you will be mathematically called to account on the small print of the licence rules. The consequences could be serious. If an unexpected bill comes, do not throw in the towel and immediately accept the offer to settle the case quickly and 'favourably', only to end up with a purchase of yet another costly product that you do not understand.

Investigate whether the results are correct: is there an appeal against the applicable agreements, is there room for interpretation, is it possible to rely on legal user rights if the supplier does not settle with you, is the result a flagrant breach of existing laws. Maybe it is about time to have another conversation with the supplier.

Judica Krikke is a lawyer, partner, and head of IP and TMT practice group at Stibbe. She is also an arbitrator at the Foundation for the Settlement of Automation (SGOA).

This article was published in Dutch in the third edition (volume 2) of online magazine Growth Quarterly in 2015.



Judica Krikke
Partner, Head of TMT/IP
Amsterdam
T: +31 20 546 02 12
E: judica.krikke@stibbe.com