intug | empowering business communications

INTUG

Proposal for a

Software Publishers' Code of Conduct

April 2016

1 Why this proposal for a Code of Conduct?

In today's fast moving software world, companies and public services share an increasing unease when it comes to managing their software assets. Software publishers release software products supported by sophisticated and complex license models and agreements. Furthermore, they seem to be either unable or unwilling to provide their customers with proper controls to protect them from using software in a manner not in line with these complex license specifications.

Typically, software publishers label software poorly and customers are not provided with accurate means and guidance to proactively manage their software estate, nor are they given the required clarity and transparency for the level of software licensing programs governing the use of the licensed software.

In a lot of cases customers accept terms and conditions to contracts in which the embedded (future) license consequences are unclear and consequently not properly understood. As a consequence, many software customers have been poor at controlling the deployment of software and have found difficulty in managing their software audit functions.

In a lot of cases software publishers have been using this lack of clarity over software licensing to their advantage in sales processes, contract negotiations and compliance reviews. This has resulted in general market dissatisfaction and distrust.

This proposal for a code of conduct defines a set of acceptable practices for software publishers' behaviour and covers contractual, technical, lifecycle and software audit conduct rules. The ultimate goal of this proposal for a code of conduct is for software publishers to provide legal certainty and unambiguous software licensing contracts with a clear definition of responsibilities for both software publishers and customers.

The 21st century will be known as the digital era. Already today, we see exponential technology growth transforming every sector. IT technology has become an important production factor for economic progress and development. Complex, unbalanced and ambiguous software license contracts are a roadblock to unleashing the full potential of digitalization.

INTUG and its members urge software publishers to accept following principles and behave accordingly.

This document was prepared in cooperation with Erik Valgaeren, attorney at Brussels and at the New York Bar and partner in the TMT group of Stibbe.

2 Guiding Principles

· Software publishers have the right to protect their intellectual property rights.

- Both software publishers and customers have the responsibility and obligation to adhere to the licensing contracts, on the assumption that these are clear and unambiguous.
- Software publishers and customers have the right to transparency, professionalism, openness and clarity throughout the complete software audit process.

3 Contract Related Guidelines

1. Software publishers should provide easy to use mechanisms for finding and filing contractual documents relevant to a specific software licensing case, in a secure and confidential way. These documents have to be provided in the version applicable to the sale in question and should include the case specific terms and conditions.
2. Software publishers have to be able to present the customer with a complete, accurate and up-to-date list of entitlements for using software produced by the publisher. This list should be provided in response to the first demand of the customer.
3. The software publisher is responsible for properly tracking and documenting the customer's software and service file. The customer is responsible for implementing proper software license contract management.
4. All contract related documentation has to make use of clear, unambiguous language that is easily understood by non-specialists. The documents need to have a clear structure and leave as little room for interpretation as possible. The documents should make limited use of specialised terminology. If specialised terminology is used the terms should be explained in a glossary to be included by the software publisher.
5. Software licensing contracts should not include clauses allowing the software publisher to impose unilateral changes to the contract and/or related terms and conditions.
6. Software publishers should provide and use localised contracts and supporting documents. Use of Anglo-Saxon terminology and legal concepts is not appropriate for customers not located in an Anglo-Saxon country.
7. Software licensing contracts should be very specific and detailed as to what kind of access is covered by the software license. The current IT environment is complex and this should be reflected in the amount of detail included in the software licensing contracts.
8. Support and maintenance of products should not be automatically linked to the use of a software product. The exact scope of support and maintenance coverage should be described in detail in the applicable contracts.
9. Addition of a new product to an already licensed software stack or maintenance renewal should be possible without changing the terms and conditions of the already licensed products in the software stack.
10. Software vendors should offer more flexible license schemes to provide for limited use of their software for
    a. business continuity and disaster recovery;
    b. reporting and auditing purposes when the software isn't used anymore. In the same line of reasoning, software publishers should facilitate the reuse and/or exportation of the data formerly managed by their software.

11. The pricing of redundancy features is in most cases unreasonably high. Software license models should not lead to the cancellation of redundancy projects hurting business continuity.

12. Contracts should not refer to web links.

4 Technical Guidelines

1. The software solution offered has to correspond to the needs as expressed by the customer.
2. The software publisher should protect the customer from accidental activation of non-free options and extension by end users. It is the customer's responsibility to associate access rights and privileges.
3. Non-free options should never be enabled by default in a standard installation of a software product.
4. Software publishers should put a program in place to certify third party tools for managing software license usage. Software asset management tools have to respect privacy, confidentiality and integrity and should in no way allow customer data or information regarding other products to be shared with the software publisher without the customers' explicit consent.
5. The software publisher is fully liable for the tool it provides to the customer.
6. Software asset management tools should make maximum use of open standards. The results generated by these tools should be recognized and accepted by the software publishers, also in the context of an audit.
7. There should be complete transparency regarding the lifecycle of products including name changes and such. The introduction of new products or versions should not be used to force the customer to migrate to new versions or products offering functions not wanted/needed by the customer and/or leading to changed contract terms and conditions and/or pricing, including support contracts.
8. Installation of software updates or patches should never lead to an increase of the license pricing as long as no additional functionality is enabled and/or used by the customer.
9. Software publishers are urged to offer license price differentiation between non-production environments such as development, backup, test and acceptance environments and production environments.
10. Software licenses should be linked to a functional scope making the usage right independent of the underlying platform and its scalability. Migration of software to a new platform should not lead to new software contracts and additional expenses as long as the functional profile of the software is not extended or altered. This would ease migration of the licensed functions to new platforms and technologies without additional negotiations, contracts and cost.
11. At the time of license sale and based on a customer request, the software publisher should sign off the implementation of its software 'as compliant' in the customer's overall architecture, including databases and virtualization, redundancy and disaster recovery setup, to avoid any misunderstanding or confusion about the proper application of the license.

12. Software licensing contracts should be very specific and transparent in the definition of the software metrics supported by the product. The transition to a different metric should be possible without extra cost as long as no additional functionality is activated or used.
13. Software publishers should provide detailed examples of how their licensing agreements make use of the supported license metrics. These examples should be based on typical, real-world cases and should provide detailed explanations on how to calculate the licensing requirements for the different license metrics supported in combination with the applicable virtualisation and redundancy features.

5 IMAC[1] Related Guidelines

1. Software license contracts should include a detailed description of the possibilities in transferring software licenses to other geographical or legal entities within the corporation.
2. Given the frequency of mergers, acquisitions and divestments software license contracts should also include a comprehensive description how software licenses can be transferred and/or consolidated in these scenarios.
3. Software license agreements and support contracts should include provisions for decreasing license and/or support fees as the use of a software product decreases.

6 Software Audit Related Guidelines

1. The software publisher initiated software audit process needs to be detailed in full transparency at the latest at the start of the audit.
2. The audit process needs to include gates for the typical milestones that characterise a publisher initiated software audit.
3. The software customer is entitled to ask for a software license audit to be performed by a neutral entity agreed upon by both.
4. Software audits are known to cover time periods going far back in the past. The time period covered by an audit should be limited in time and needs to be reasonable.
5. At all times the customer should be allowed to veto the use of publisher provided data collection tools.
6. The software licensing contract needs to be very clear about what data the customer is to hand over to the software publisher in case of a publisher initiated software audit. The data handed over to the software publisher can be used only within the framework of the specific software audit and should never be disclosed to other parties. When appropriate a non-disclosure agreement needs to be put in place in case of an audit.
7. Software audit procedures based on the use of raw data and non-transparent post-processing by the auditor and/or software publisher are unacceptable. The auditor and/or software publisher have to fully disclose the methods used for collecting both the software use and the entitlement data sets and it is the software publisher's responsibility to provide convincing evidence for the alleged discrepancies.

[1] Install, move, add and change

8. When determining the settlement fee of a software audit, software publishers should also take detected cases of over licensing into account
9. The demonstrated good faith of the customer in the performance of the contract should be taken into account. When the customer accepts the use of unpaid licenses but where its good faith is recognized, the customer should be allowed to pay the licenses at the discounted price of the agreement customer currently has with the publisher (and not at the public price).
10. The situation after the software audit and the rightsizing should constitute a new origin from a software licensing point of view. Under no circumstances it is acceptable for future software audits to reconsider software usage in the period preceding the new origin. This origin date has to be defined explicitly in the settlement letter.
11. Back-charging for software considered unlicensed is only acceptable for effectively executed activities, typically maintenance and/or support, within the scope of the unlicensed software.
12. Software license contracts should clearly state which party will bear what part of the cost of the software audit. Each party bearing the cost of its resources used to support the software audit would achieve a fair division of the cost.
13. In the context of publisher initiated audits software publishers should refrain from threatening or intimidating their customers. This includes threats to suspend the use of software and/or threats of initiating legal action.
14. In the context of the settlement of a software audit it is not acceptable to put the customer under time pressure to accept the settlement proposal.