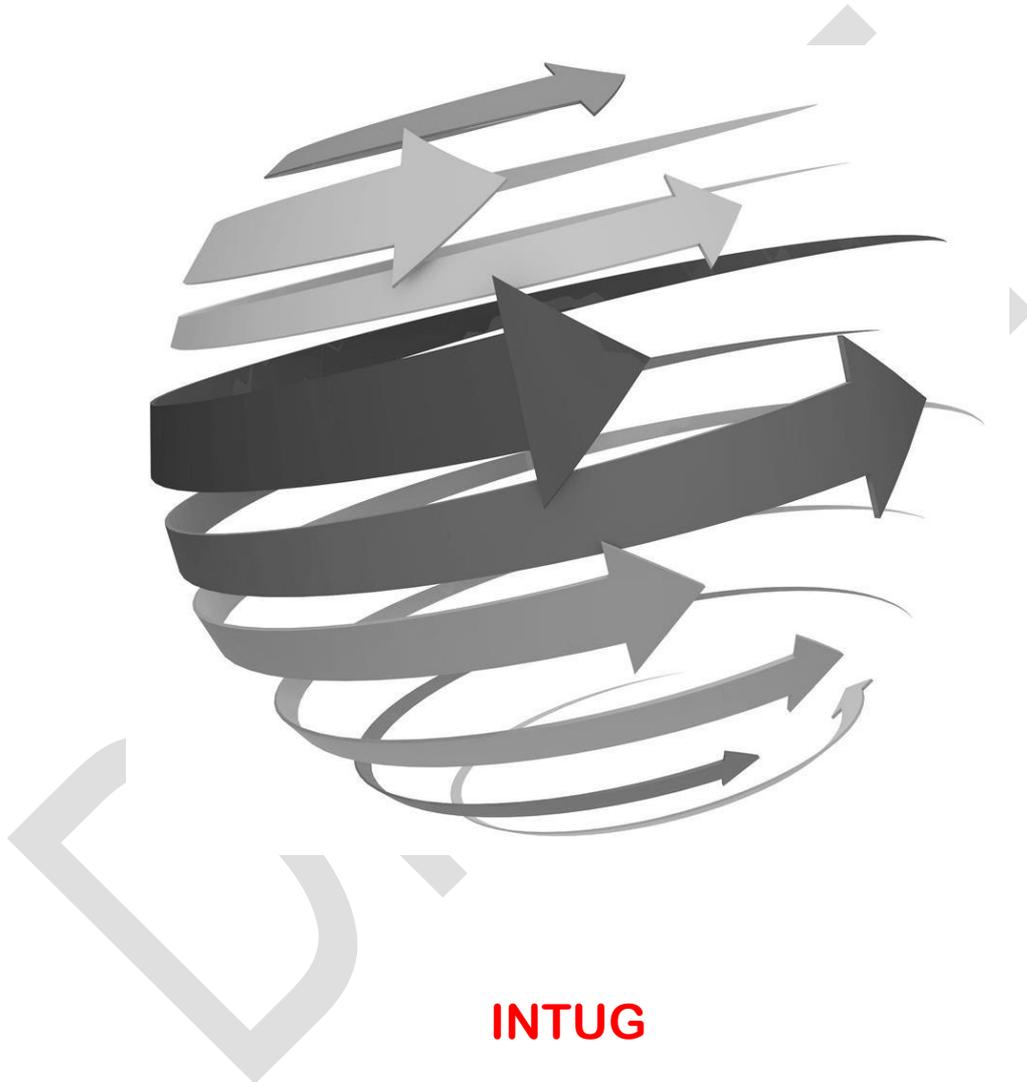




intug | empowering business communications



INTUG

Proposal for a
Software Publishers Code of Conduct

September 2015

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 vendors 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 avoid that customers use software in a manner not in line with these complex license specifications.

Typically, the software publishers poorly label software and the customers are not provided accurate tools and guidance to proactively manage their software estate, nor are they given the required clarity and transparency on 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 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 publisher's 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 vendors to provide legal certainty and unambiguous software licensing contracts with a clear definition of responsibilities for both software vendors and customers.

The 21st century will be known as the digital era. Already today, we see exponential technology growth transforming every sector or industry. 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 the digitalization of our society holds.

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

2 Guiding Principles

- Software publishers have the right to protect their intellectual property rights.
- Software publishers have the right to exercise clauses integrated in software licensing contracts.

- Both software publishers and customers have the responsibility and obligation to adhere to the clauses as stated in the licensing contracts.
- Customers have the right to deny audit requests that are not the result of contractual obligations or evidence based breaches of intellectual property rights.
- 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 vendors 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 vendors 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 vendor. This list should be provided upon first demand of the customer.
3. The software vendor 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 vendor.
5. Software licensing contracts should not include clauses allowing the software vendor to impose unilateral changes to the contract and/or related terms and conditions.
6. Software vendors 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 (direct/indirect) is covered by the software license. The current ICT 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 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 business continuity, reporting and auditing purposes after stopping use of that software in a production context. In the same line of reasoning, software vendors should facilitate the reuse and/or exportation of the data formerly managed by their software.



4 Technical Guidelines

1. The software solution offered has to correspond to the needs as expressed by the customer.
2. The software vendor is responsible for allowing maximum use of the technical means (e.g. role-based access control, privileges and access rights) to prevent the accidental activation of non-free options and extension by unauthorized persons. It is the customer's responsibility to associate access rights and privileges to physical users.
3. Non-free options should never be enabled by default in a standard installation of a software product.
4. As a general guideline the use of software vendor supplied tools during audits and for managing license usage is unacceptable. Software vendors should put a program in place to certify third party tools for managing software license usage, preferable open source. These tools should be well documented, describing in detail the audit mechanism that will be applied, and which information will be revealed to the software publisher.
5. Software asset management tools should make a maximum use of open standards. Software license management should be done using transparent data and tools, the results of which are officially recognized and accepted by the software vendors.
6. Software asset management tools should allow the customer to limit the information presented to a software vendor. These tools have to respect privacy, confidentiality and integrity and should in no way allow information regarding competitive products to be passed to the software vendor or auditor.
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 actively enabled and/or used by the customer.
9. Software vendors are urged to offer license price differentiation between non-production environments such as development, 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. 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 vendor will sign off the implementation of his software 'as compliant' in an 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. The pricing of redundancy features is in most cases unreasonably high. Software licence models should not lead to the cancellation of redundancy projects hurting business continuity.
13. 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.
14. Software vendors 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 IMAC¹ Related Guidelines

1. Software license contracts have to 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 on how software licenses can be transferred 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. In most current license agreements there are only provisions for increasing the license and/or support fees to be paid by the customer.

6 Software Audit Related Guidelines

1. The software vendor initiated software audit process needs to be detailed in full transparency in the software licensing terms and conditions.
2. The software licensing terms and conditions also need to include gates for the typical milestones that characterise a vendor initiated software audit. The audit process cannot continue past a defined gate as long as both parties do not agree with the results of the actions preceding a gate.
3. The software customer is entitled for a software license audit to be performed by a neutral entity agreed upon by both. The software licensing terms and conditions should thus clearly define which organisations/companies are allowed to perform a software vendor initiated audit and which parties are explicitly denied from executing the software audit because of for example conflicting interests.
4. The license agreement needs to be very clear on what time period a software license audit is allowed to cover. Software audits have been known to cover time periods going far back in the past.

¹ Install, move, add and change



5. The use of software vendor provided data collection tools is unacceptable unless these tools have been verified and approved by a neutral entity. At all times the customer should be allowed to veto the use of vendor 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 vendor in case of a vendor initiated software audit. The data handed over to the software vendor can be used only within the framework of the specific software audit and should never be disclosed to other parties.
7. Software audit procedures based on the use of raw data and non-transparent post-processing by the auditor and/or software vendor are unacceptable. The auditor and/or software vendor have to fully disclose the methods used for collecting both the software use and the entitlement data sets and it is the software vendor's responsibility to provide convincing evidence for the alleged discrepancies.
8. In the settlement phase of a software audit, software vendors should also take detected cases of overlicensing into account when determining the settlement fee.
9. 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 that future software audits would be allowed to reconsider software usage in the period preceding the new origin. This origin date has to be defined explicitly in the settlement letter.
10. 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.
11. 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.
12. In the context of vendor initiated audits software vendors should refrain from threatening or intimidating their customers. This includes threats to suspend the use of software and/or threats of initiating legal action.
13. 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.

Copyright © INTUG/BELTUG 2015. This document may be freely duplicated and distributed in its entirety on an individual one to one basis, either electronically or in hard copy form so long as it is attributed to INTUG/BELTUG. It may not, however, be disassembled or modified in any way as part of the duplication process.