CONFIDENCE FOR THE DIGITAL AGE
SOFTWARE RISK MANAGEMENT

Vol. 2, No. 3 - July 2002

Legal Perspectives
UCITA and Software Licensing
By Geoffrey T. Hervey, Esquire

A new model law—the Uniform Computer Information Transactions Act, or UCITA—is puzzling software vendors and users alike. UCITA is controversial and has generated substantial publicity, both positive and negative. This article does not attempt to weigh UCITA's relative good and bad points, but merely explains certain provisions of the Act in the broader context of software licensing. While much of the criticism of UCITA centers on provisions dealing with consumers (e.g., individuals obtaining software for home and family use), this article focuses on commercial licensors and licensees. In addition, the article highlights certain UCITA features that are important to those involved in the management of the risks associated with software failure.

First, A Brief History of UCITA

Drafted by a panel of scholars, lawyers and other experts formed as part of the National Conference of Commissioners of Uniform State Laws (NCCUSL), UCITA is not an actual law but merely a model law that the various states may or may not ever adopt. In fact, only two states—Maryland and Virginia—have enacted UCITA as a law and it is

Recommend this article to a friend

Printer-friendly version

under consideration in only a handful of other states. Nevertheless, UCITA is important to software vendors and users nationwide because, even if it is not widely enacted into law, it provides valuable guidance on critical issues involved in any software licensing arrangement.

In drafting UCITA, the NCCUSL panel sought to bring together the various legal and practical rules that govern the licensing of software around the country. Since no law was specifically directed to software licensing, different courts in different parts of the country applied different laws and rationales to software licensing arrangements[1]. This is still largely the case. For example, in some jurisdictions software is considered to be "goods" governed by the Uniform Commercial Code and other laws covering the sale of other goods, like refrigerators and "widgets." Other jurisdictions, however, apply different laws that focus more on the intangible aspects of software. This can make it difficult to assess rights, obligations, and remedies under licensing arrangements from state to state.

Software Licensing Considerations

- **Written Agreement:** The parties to a commercial software license arrangement should commit their agreement to writing. Both sides have an interest in ensuring that the terms of the deal and are clearly set forth to prevent unfounded expectations and minimize disagreements later.
- **Warranties Included and Disclaimed:** Both parties must be satisfied with the warranties provided and disclaimed. Typically, a customer's only remedy will be for breach of warranty. The customer should ensure that the agreement clearly explains included warranties. The licensor should make sure that any unintended warranties, especially implied warranties, are disclaimed.
- **Remedies and Limitations of Liability and Damages:** The contract should explain the remedies available to the customer if the software does not work as promised, as well as any limitations on the types of liability that can be imposed on the licensor (*e.g.*, for lost profits) and any dollar limits (such as not more that the license fees paid).
- **Acceptance Criteria:** The concept of "acceptance" of the software by the

customer is crucial to both parties because the beginning of warranty periods and payment obligations are generally tied to acceptance, and acceptance has revenue recognition implications for the licensor. The contract should state the exact conditions that constitute the customer's acceptance of the software.

- **Miscellaneous Provisions:** Parties to a commercial software licensing agreement should agree on a number of smaller issues, including governing law, whether lawsuits can be filed only in a certain state or court, whether the parties must arbitrate any dispute, whether the prevailing party can recovery its attorney fees, and whether the parties can publicize their agreement.

Rather than try to re-write existing law, the NCCUSL panel attempted to create a set of rules that reflect how software is actually licensed in the real world. (In this writer's opinion, the panel largely succeeded in doing so.) Moreover, UCITA consists largely of default provisions that dictate how a software licensing agreement is to be interpreted when the parties have not agreed upon specific provisions. In fact, parties to a commercial agreement can agree that UCITA does not apply at all, that is, they can "opt out" of the Act altogether.

UCITA covers not only software agreements, but also just about any agreement that involves the transfer of code, electronic information, electronic data or other bits and bytes. For example, UCITA covers information databases in electronic form, online access agreements and other "transactions" involving "computer information" (which is a broad term that UCITA defines to include software programs, data, text, sound images and other information that is "in electronic form which is obtained from or through the use of a computer or which is in a form capable of being processed by a computer").

Which State's Law Applies?

The first issue to consider when assessing UCITA's impact on software licensing is which state's version of the law applies. As previously stated, UCITA has been enacted as law only in Maryland and Virginia. If a software licensing contract provides that the law of another state applies, or if it is silent on this point and the contracting parties are not in Maryland or Virginia, then UCITA is not an issue (at least as of the date of this writing). Yet, since UCITA reflects the real-world practicalities of software licensing, it

provides an excellent outline of issues to consider in any software licensing arrangement.

Is There a Written Agreement?

This article assumes that a licensing arrangement is covered by a written software licensing agreement. The need for a written agreement may be self-evident to lawyers, but may not be so obvious to others.

The main purpose of a written agreement is to ensure that the parties understand all of the terms of the deal to minimize the chances of a dispute—or a lawsuit—down the road if unwritten expectations are not met. This is especially so with respect to defining what is being licensed and what it is supposed to do. The agreement should clearly state what program (and version) is being licensed and contain a description of what it is supposed to do. Sometimes the parties may attach the program's specifications to the contract as an exhibit, and the licensor may warrant that the program will perform in accordance with its specifications. Such specificity should reduce unfounded expectations of the program's capabilities, and a realistic description of those capabilities should improve the chances of satisfying the customer. A well-drafted agreement will also set out the parties' rights and obligations, and any limitations on them. Moreover, if a written agreement does not exist, UCITA's default provisions will apply in those states in which UCITA is the law. The better course is to have a written agreement in place.

Understanding UCITA's Various Warranties

UCITA specifies numerous warranties that a licensor must make to a licensee, most of which can be disclaimed in the contract. Obviously, the parties to a software licensing agreement have competing interests in asserting and disclaiming warranties. Moreover, different warranties can arise depending upon whether the software in question is an off-the-shelf product or is built to a customer's specifications (*i.e.*, custom software) and what other types of information are involved, such as brochures, demonstrations and advertising. The following is a discussion of some of the more significant warranties provided by UCITA[2]:

Non-Interference: The user of a software program always faces the

possibility that a third party may claim that the user has no right to use the program because the software was misappropriated or violates the third party's rights, such as a copyright. The user may be also forced to stop using the software, which could be detrimental if the software is mission-critical to the user's business. UCITA provides that a licensor warrants that the software will be free of the rightful claim of a third party.

(Notably, this warranty may not apply with respect to custom software. If a licensee provides a licensor with detailed specifications for a software application and tells the licensor how to meet those specifications, then a non-interference warranty is not provided by UCITA. Further, UCITA specifically states that the licensee agrees to hold the licensor harmless with respect to any claim made by a third party that results from the licensor's compliance with the specifications or instructions. These provisions, however, would seem to apply only when the customer provides extremely specific instructions for highly specialized, custom software, and not to mere customization of otherwise generally available programs.)

Express Warranties: An express warranty is a statement guaranteeing that the software will perform a specific function (*e.g.,* process X number of records in a set period or run on a certain operating system). If the software does not perform consistently, the express warranty is breached, entitling the customer to demand that the licensor correct the error and/or pay damages, unless damages are limited or excluded (which, as explained below, is normally the case).

Implied Warranties: Unlike express warranties, implied warranties arise not from statements made about software but from the circumstances surrounding the license. The breach of an implied warranty will require a licensor to effectuate a remedy and/or pay damages, depending on the limitations in the contract. Some of the more common implied warranties

include:

- **Implied Warranty of Merchantability:** This implied warranty means that a software program is fit for the ordinary purposes for which programs of that type are used. Notably, however, this does not mean that the software is the best available or that it is fit for all possible uses. Moreover, UCITA recognizes that all software programs contain bugs and flaws. Therefore, the implied warranty of merchantability under UCITA requires only that the program be generally within the average standards applicable in the industry for similar programs. The implied warranty does extend, however, to any promises or affirmations of fact that are made on the container or label. Most licensors will specifically disclaim this implied warranty.
- **Implied Warranty That the Software is Fit for the Licensee's Purposes:** This implied warranty arises when, at the time of contracting, a licensor has reason to know of a particular purpose for which the particular licensee intends to use the program. UCITA provides that, under these circumstances, the licensor impliedly warrants that the program will be fit for that particular purpose.
- **Implied Warranties That May Arise from Conduct:** Warranties do not arise solely from the contract or the circumstances. Sometimes, implied warranties can arise from the manner in which the parties deal with each other or from the nature of a particular industry. Such warranties can arise by implication if the parties or the industry as a whole treats certain warranties as if they exist. For this reason, contracts typically include a statement disclaiming any such warranties.

General Disclaimers of All Warranties: UCITA permits parties to disclaim warranties with a general disclaimer rather than specifically disclaiming every possible warranty. A typical disclaimer might read as

follows:

> EXCEPT FOR WARRANTIES EXPRESSLY SET FORTH IN THIS AGREEMENT, ANY AND ALL LICENSED MATERIALS ARE DELIVERED "AS IS" AND THE LICENSOR MAKES AND THE CUSTOMER RECEIVES NO ADDITIONAL EXPRESS OR IMPLIED WARRANTIES. THE LICENSOR HEREBY EXPRESSLY DISCLAIMS ANY AND ALL OTHER WARRANTIES OF ANY KIND OR NATURE CONCERNING THE LICENSED MATERIALS, WHETHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF TITLE, MERCHANTABILITY, QUALITY, ACCURACY, OR FITNESS FOR A PARTICULAR PURPOSE OR THE CUSTOMER'S PURPOSE. THE LICENSOR EXPRESSLY DISCLAIMS ANY WARRANTIES THAT MAY BE IMPLIED FROM USAGE OF TRADE, COURSE OF DEALING, OR COURSE OF PERFORMANCE. EXCEPT FOR THE EXPRESS WARRANTIES STATED IN THIS AGREEMENT, THE LICENSED MATERIALS ARE PROVIDED WITH ALL FAULTS, AND THE ENTIRE RISK OF SATISFACTORY QUALITY, PERFORMANCE, ACCURACY, AND EFFORT IS WITH CUSTOMER. THE LICENSOR DOES NOT WARRANT THAT THE LICENSED MATERIALS WILL OPERATE WITHOUT INTERRUPTION OR BE ERROR FREE.[3]

These clauses are, for the most part, enforceable. If such a clause is included in a contract, the parties will only be able to assert those warranties that are specifically set out in the contract and not disclaimed.

Acceptance and Warranties

The concept of "acceptance" of the software by the licensee has an impact on warranties, remedies and other aspects of the transaction (including when payments are due and whether and when the licensor can recognize the revenue from the license). In most cases, the beginning of the warranty period is tied to the acceptance date. The licensee may assume that the program is not accepted until it is installed and tested in a production environment, while the licensor may deem the program to be accepted when the program materials are delivered to the licensee. The agreement should clarify when acceptance occurs. If the contract does not specify clearly the time at which acceptance occurs, UCITA provides only generalized guidance, holding that acceptance occurs when, for example, the customer acts in a manner that is inconsistent with non-acceptance, or obtains substantial benefit from the program. Because these are vague, subjective terms, they invite dispute and litigation. The better course is to avoid such problems by including acceptance criteria in the agreement.

"Obviously, the parties to a software licensing agreement have competing interests in asserting and disclaiming warranties."

Remedies and Damages

UCITA provides remedies when one party breaches a software licensing agreement and specifies the damages that can be collected. Reflecting commercial reality, however, most remedy and damages provisions in UCITA can be overruled by contract terms that limit remedies and damages. Further, most software licensing agreements expressly limit remedies to repair or replacement of the software and limit damages to the amount paid for the software. Such clauses are enforceable under UCITA.

Remedies

As a general matter, UCITA provides that a party aggrieved by the other party's breach of contract "has the remedies provided in the contract or in [UCITA]." UCITA provides that the customary remedies for breach of contract are not displaced. This includes suing for damages and for equitable remedies, such as a suit asking the court to force the breaching party to perform (called "specific performance"). UCITA also provides that an aggrieved party can cancel the contract for breach. Cancellation essentially treats the

contract as over and excuses any further performance under the contract. Cancellation is only available, however, for a material breach of the contract, and a breach is material if:

- the contract provides that the type of breach is material;
- the breach involves a substantial failure to perform an essential element of the contract; or
- the circumstances (including the language of the contract, custom in the industry, and other factors) indicate that the aggrieved party suffered substantial harm or the breach deprived the aggrieved party of the benefit of the contract.

Establishing that a breach is material can be difficult. For that reason, a prudent party will specify in the contract that a breach of certain provisions is or is not a material breach. If the contract is cancelled, however, the parties do not simply walk away. UCITA contains numerous provisions dictating what happens upon cancellation, including whether the licensee may continue to retain and use the software. Notably, UCITA provides that the parties may agree that the contract may not be cancelled at all.

Despite the fact that UCITA recognizes various remedies, it also allows the parties to limit the remedies that are available for breach. For example, a software license agreement might limit the aggrieved party's remedy to the replacement of the software or a refund of fees paid for the software. It should also be noted that, generally speaking, a party found to be in breach must be given a chance to "cure" the breach. Therefore, the contract may contain clauses directing the parties to give each other notice of potential breach conditions and requiring that the notice be in writing and that it contain a detailed description of the problem so that the other party can attempt to fix the breach.

Damages

Damages fall into several categories. "Direct damages," for purposes of this article, essentially constitute the price paid for the software and do not include other damages that were caused by the software's failure. Such damages are called "consequential damages," and include lost profits and royalties, damage to reputation, and the loss of anticipated benefits from the planned use of the software. UCITA, as a general matter, allows recovery of direct and consequential damages. UCITA also recognizes the

concept of "cover," that is, the need to purchase a different program to replace the defective one. The difference in price between the failed software and the cover program may be a recoverable damage if the choice of the cover program was commercially reasonable.[4]

Notably, UCITA specifies that damages cannot be recovered if the aggrieved party could have avoided them by taking reasonable measures. In addition, damages can only be recovered if the aggrieved party can demonstrate, first, that a specific contract provision—such as a warranty—was violated and, second, that the breach of that provision was the proximate cause of the damages. For this reason, warranty disclaimers have the effect of making damages unavailable with respect to the disclaimed warranty.

Significantly, UCITA permits the parties to limit the types of recoverable damages, and almost all commercial software licensing agreements contain a limitation of damages (or "limitation of liability") clause. Such clauses are generally enforceable, at least in a commercial transaction. A typical clause will read as follows:

> IN NO EVENT WILL THE LICENSOR BE LIABLE FOR ANY INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES, INCLUDING LOSS OF PROFITS, LOSS OF REVENUES, LOSS OF DATA, LOSS OF USE, ANY OTHER ECONOMIC ADVANTAGE OR COST OF COVER INCURRED BY CUSTOMER ARISING OUT OF THIS AGREEMENT, UNDER ANY THEORY OF LIABILITY, WHETHER IN AN ACTION IN CONTRACT, STRICT LIABILITY, TORT (INCLUDING NEGLIGENCE) OR OTHER LEGAL OR EQUITABLE THEORY.

Similarly, UCITA permits the parties to limit not only the type of damages that may be recovered but also the amount of those damages. Frequently, software agreements will limit damages to the amount paid for the software. If the software consists of modules or components, the contract may limit damages to the fees paid for the specific module that failed. Moreover, some contracts prorate damages over a specific period of time and allow the licensee to recover only a portion of the fees paid, prorated over that period.

Other contracts limit damages to the amount paid or to a set limit, whichever is less. Such clauses are generally enforceable.

Conclusion

Although UCITA has been enacted into law only in Maryland and Virginia, it was designed to reflect the "state of the industry" nationwide. And while none of the UCITA provisions discussed above significantly alters existing commercial practice regarding the licensing of software, the Act validates many common practices and clauses and places them in a meaningful and organized framework. UCITA is most useful in this regard in that it constitutes a realistic checklist of items of concern to both licensors and licensees of software.

The lesson to be drawn from a review of UCITA is that anyone reviewing a software license agreement from a risk-management perspective needs to pay close attention to the language addressing warranties, remedies and damages.

ENDNOTES:

1. Software is "licensed" rather than "sold" because it is a creative work that is covered by copyright laws. As such, once it is "sold," the seller gives up all right to the work, including the right to "sell" it to someone else. A developer thus needs to "license" use of the software to the end user, while retaining legal ownership of the program, so that he or she can license it over and over to many users and collect fees. In the same way, music recordings are not sold but licensed, even though one "buys" a music CD. Napster ran into trouble because it allowed people to make and give away copies of music that they did not own but had only licensed.
2. This discussion assumes that the software licensor is a software merchant, that is, one who routinely develops and/or licenses software as a business. Different rules apply to non-merchants and are not discussed here. Moreover, this article does not address all of the warranties provided under UCITA or other laws.

3. Because UCITA and other laws require such general disclaimer clauses to be conspicuous, they are frequently printed in all capital letters, or in bold or some other contrasting font.
4. Other types of damages may also be available. For example, "incidental damages," which include shipping costs, etc., are sometimes awardable. "Punitive damages," which are designed to punish intentionally wrongful conduct, are almost never available in a straight breach of contract action. This article does not address all damages that may be available.

Geoffrey T. Hervey, Esq. (ghervey@bregmanlaw.com) is a partner in the Bethesda, Md. law firm of Bregman, Berbert, Schwartz & Gilday, LLC (http://www.bregmanlaw.com/).