In the world of contract law, “good fences make good neighbors.”¹ In other words, a well-written contract details the expectations of both parties; it is not supposed to help win a lawsuit, it is to prevent a lawsuit from ever occurring. *The Tech Contracts Handbook*, by David W. Tollen, tries to assist business people and lawyers who are unfamiliar with the technology field with drafting basic contracts involving software licenses, software services, and combinations of the two.² Although this book is meant to act as a guide to drafting these contracts, the author cautions that, when in doubt get a lawyer, or if the reader is already a lawyer, get a colleague that has expertise in technology contracts. Tollen organizes the book by the types of clauses found in a software contract; transactional clauses, general clauses, and supporting clauses. The author discusses these clauses in the framework of the issues surrounding software licenses and software service contracts.

Before diving into the nuts and bolts of drafting a software contract, the author makes a few points about intellectual property (IP) to assist a layperson, who has no experience in IP, with understanding the nature of the rights that are transferred in a software license. As software and computer programs are usually considered to be a form of literary work³ and capable of

² The author, David W. Tollen graduated with honors from Harvard Law School and cofounded Adeli and Tollen LLP, where he specializes in technology and intellectual property transactions.
copyright protection, it is important to understand the restrictions that copyright law places on ownership rights and software licenses. The author notes that although a copyright will protect the owner against infringement of the actual written code of software, it will not prevent someone from taking that program and altering it so as to make a new program or code.

As copyrights are not absolute in protecting a software provider’s proprietary interests, the licensing agreement must strike a balance between allowing the receiver of the license meaningful use of the software, while restricting the use to not jeopardize the provider’s IP interest. Striking this balance between a receiver’s rights and a provider’s copyright interest is not based on a notion of fairness, but on leverage during negotiation. The contracting party who has more leverage can set better terms and it is imperative that the contract accurately reflects these terms, as the slightest drafting error can be disastrous for the parties’ expectations. The central idea is that it is necessary to be precise in balancing these interests so as to not jeopardize contractual expectations through poor drafting. This book attempts to aid in such drafting.

The author begins with a discussion of the clauses that are fundamental to the deal, the transactional clauses. These clauses detail the transfer of software rights and/or services and the methods of payment in return. To begin his discussion of transactional clauses, the author specifies the various methods and issues surrounding software licenses.4 This is where a balance between the copyright and the licensee’s needs should be established. The author suggests narrowing the terms with exact precision. A well-written and detailed software license, particularly in defining the scope of the use, is necessary for the user to definitively know when the license and his rights as a copyright holder are being infringed. This is a situation where

---

4 A software license involves allowing the recipient of the license to copy and use a piece of software in a limited capacity, leaving the ownership rights in the software with the provider. See TOLLEN, supra note 1, at 3. Essential to the retention of ownership rights in a software license is an explicit statement in the contract that the provider retains all ownership rights, and the licensee only obtains a license. See TOLLEN, supra note 1, at 4.
“good fences make good neighbors,” as drafting the contract in a specific manner will provide both parties with advanced notice of knowing the other’s expectations. The software provider understands exactly what they are giving up, and the recipient knows the limits of his permitted use.

The author discusses that an area of particular importance is in clauses describing a recipients distribution rights. When allowing for distribution of software, the third party use and distribution are secondary considerations that need to be accounted for by the provider. A third party is not a party to the contract and therefore does not have that notice in writing about what type of use is permitted. To address this issue the author suggests limiting the scope of the distribution, through allowing for termination of the license if the recipient fails to enter into contracts with third parties to limit their use and distribution. From this, the provider can have assurance of a remedy if the recipient does not restrict the use of third parties, and the balance between use and copyright are maintained.

Formulating and drafting an agreement that fits the expectations of the parties can be difficult when speaking about intangibles such as software code and services. The author suggests that in defining these key provisions technologically sophisticated parties may be necessary as some lawyers are not properly equipped with the technical know-how to capture those expectations in writing. The difficulty from a legal practice or business perspective is knowing when to defer to those that have more expertise. The author suggests that it is better to be safe than sorry, and that when in doubt to get a lawyer. However, what may be more difficult is knowing where that threshold for doubt lies. The book does not address this issue, and it may be that providing this kind of guidance is so dependent on the circumstances as to be impossible.

5 TOLLEN, supra note 1, at xiv.
to advise upon. With this thought it mind, it may remind the headstrong attorney or businessman to be hyper-vigilant not just to the terms of the contract, but to personal capacities.

In part II the author discusses issues surrounding drafting general clauses of the contract, which are those clauses that are supplementary to the main agreement but are not part of the contracts boilerplate. Although these clauses are not the essence of the bargain, they usually cause the most disputes. The clauses often involve the technical specifications of the contract that, as the author notes, should usually be drafted under the supervision of an IT staffer. This is usually agreed to in advance and is usually not an issue. However, general clauses also include schedules, delivery, and upgrades to the software, and, in a field that is ever-changing, these clauses become crucial.

The utility and time sensitive nature of such software is also crucial to the recipient, making the source code one of particular importance to them. If the provider breaches, the recipient is left without the software it has contracted for. To remedy this and protect the recipient, the author suggests terms requiring the provider to place the source code in escrow. Having the material placed in escrow protects the recipient of the software license in the event that the provider breaches the contract or goes out of business. This way the recipient will be able to get the source code and allow another party to use that code to salvage their expectations. This requires additional provisions for verification, maintenance of confidentiality, and identification of events triggering the release the source code. The provider at the same time understands fully that in the event of her breach, that content can be transferred to the recipient, and through the slightest modification, the value in the copyright. The provider inversely should

---

6 See TOLLEN, supra note 1, at 55.
7 See TOLLEN, supra note 1, at 70. Source code is the code that is the original version of software. See TOLLEN, supra note 1, at 256.
protect itself from disclosures of its software that destroy value in the copyright through drafting a confidentiality clause.

Although the author suggests that contractual terms will be enforced as written, this may not always be the case. In certain instances, courts will not enforce a specific term if it is against public policy, therefore knowing this limitation is essential to ensuring enforcement of the agreement. Limitations of liability and non-competition clauses demonstrate these legal quagmires. Limitations of liability are contractual terms stating that in the event of a breach one party, usually the provider, will not be responsible for certain damages.⁸ Court’s are skeptical of these provisions and can strike them from the contract. At worst, they may void the entire contract if it is deemed unconscionable. The problem is that this doctrine does not provide valuable guidance in drafting the agreement. The author suggests drafting a clause that is as forward as possible so as to create notice but not make it so restrictive as to deprive of a meaningful remedy. To add further protection, the author suggests adding language to the clause to allow survival of the limitation even if its application to a particular situation would be unconscionable.

Despite the intent to prevent disputes as to contractual terms, the provisions in this section show optimism for the capacity of business persons to focus on contingencies. When parties enter into a contract, and the proverbial exchange of promises takes place, the souring of the contractual relationship may not be quite so apparent. Focusing on contingencies is generally a calling card of lawyers, and as such the provisions in this section may be better suited for sophisticated business people or those who have confronted the complexities of contractual

⁸ These clauses usually apply to recovery of consequential damages which are those damages which are not directly attributable from the tortious conduct. See TOLLEN, supra note 1, at 110. The author uses the example of construction software, the failure of which would result in direct damages of not being able to engage in the project, and consequential damages would be those damages in lost opportunities from using the project to generate revenue.
breaches. Therefore, the emphasis placed on the importance of these terms, and the unlikely focus that these provisions will receive from inexperienced business people alters the premise that lawyers are not merely an option of convenience but a business necessity.

In the last part, the author discusses “Supporting Clauses” which he describes as the boilerplate of the contract. The author warns that although these clauses tend not to be too controversial, they are not to be taken lightly. These clauses usually involve defining various technical terms, as well as providing for recitals of consideration to ensure validity of the contract. The author suggests including a time is of the essence clause and a merger clause to ensure that the parties know exactly when a lag in the performance can amount to a material breach. This section also involves conflict clauses, which are common in instances where there are supplemental agreements to the contract. These conflict clauses elaborate on what the meaning of the contract will be when a certain document conflicts with the meaning of the actual contract. The author wraps up the “Supporting Clauses” section with a discussion of amendments and drafting a procedure to allow for future amendment to the contract.

The book finishes with an in depth discussion of the internet and emerging software issues. One such issue is open source software. Open source software allows for the recipient to have access to the source code of the software, allowing for the recipient to modify, alter, and fix certain bugs in the software. Copyleft is a form of open source software which can be devastating for the unknowing recipient, or would be distributor, of a piece of software and wishes to retain privacy of the source code. It requires all programs that incorporate certain open source software to disclose their source code. In essence, a program that uses copyleft as any part of its software is “infected” with this disclosure requirement of the incorporated code. The author suggests that those wishing to distribute or modify software without having to disclose the
source code should include certain warranties against copyleft software being incorporated into
the provided software. Although a remedy can be negotiated in the event of software being
“infected,” such remedy may be insufficient to fix the damage that the copyleft has caused.

With words like “infected,” “piracy,” and “worms” the internet for a novice business man
or attorney can be a vast and intimidating setting. What lies beyond the displayed images on a
computer screen is a mystery for a majority of people, and the idea of software code is bafflingly
complex. The doctrines invoked in copyright are hard enough to master and copyleft only shows
the laws evolution with this dauntingly sophisticated technology. The author provides a little
reassurance here, his simplification of open source, copyleft and other current issues does not
provide definitive answers, but allows for a basic building block for greater understanding to
accumulate. Ultimately it shows that with careful drafting and sufficient safeguards these risks
can be contained using contracts.

*The Tech Contracts Handbook* provides a practical guide to negotiating and drafting a
software contract. For those unfamiliar with the IT field, the book explains and illustrates
necessary principles in an easy to understand manner. As warned by the author, this is not a
replacement for a well-seasoned lawyer or even a well-versed IT professional; however, it is a
thorough overview into the nuances of an increasingly significant practice area. This book
affords the reader a greater understanding of what should be included in a software license, how
to avoid unwanted liability, and how to draft a contract that encapsulates the parties’
expectations. The topics I discussed are just small portions of the breadth of information this
book offers, and the author includes helpful exhibits, including a fully written contract for the
reader’s reference. This book is perfect for the law student who wishes to see the principles she
learned in first year contracts applied in a way that prevents the drafting problems case books are
filled with, and for these reasons I recommend this book.