Software Contract Considerations

Disclaimer: *The materials and information referenced in this document are provided for general and informational purposes only. They are not offered as, and do not constitute, legal advice or legal opinions.*

Applies to: All National Apartment Association (NAA) members who participate in the negotiation, review and execution of software contracts.

Overview: Software contracts present unique challenges for customers and are rarely the result of an "arm's length" negotiation. Instead, potential customers are confronted with the supplier's "standard" contract language. The "standard" contract was designed by the vendor's lawyer to be repeatedly used with little room to negotiate more favorable terms, much like the residential lease contracts used by property management companies. In fact, a software contract is often described as a "License Agreement," which is essentially a lease to use the software over a specified time period and under specified conditions. Before signing a contract (or recommending that someone else does), You **MUST READ IT** and familiarize yourself with the details. Below are examples of frequently encountered terms and considerations to help you understand and negotiate favorable terms in your software technology contracts.

Guidance:

Due Diligence: First and foremost, potential customers should thoroughly evaluate the potential vendor's product(s), as well as the vendor itself. For example: Does the potential vendor's product sufficiently meet your needs? Are similarly situated vendor clients successfully using the vendor's product? Does the vendor have a proven track-record over a meaningful time period? Is the vendor actively investing in developing, refining, upgrading and enhancing its product(s)? Is the potential vendor organizationally and financially stable? Does the potential vendor have the resources to adequately address your needs during the lifespan of your expected use of their product?

Entire Agreement: These provisions specify that the signed contract is what will govern your relationship. Not the pitch by the salesperson, not a response to a Request for Proposal, not a memorandum of understanding and not earlier drafts of the contract. Only the signed version will control. If there was something of importance discussed weeks or days ago, ensure it is included in the contract language.

Entities/Properties: It is important to consider whether the appropriate entity or entities are identified in the contract, and whether the ability to utilize the contracted software or services will extend broadly to directly owned, indirectly owned, jointly owned and managed and acquired properties and communities.

In addition, it should be verified whether there is any element of the contract that would require consent by a lender or joint owner. While these consent requirements are less likely to apply to software contracts than leases or easements, the question should still be considered.

Assignability and Termination: Ideally, a contract should be assignable at the owner/manager's option on a property-by-property basis and overall. Many software and service providers will resist this, particularly without their consent and, more particularly, to a competitor. Ensure the contract addresses what will happen if you sell a community subject to the contract. You should address what happens when you sell/stop managing a given community in your portfolio, stop using the software with respect to that community and cessation of future payments based on use for that community.

Term: The term of the agreement dictates how long you will be doing business with this vendor. Generally, shorter terms benefit the customer because it makes it easier to "change partners" once the term ends. However, careful consideration should be given to term length where the relevant software is core/critical to your business and/or enterprise software. Provisions that call for "automatic renewal" for the same contract term unless written notification is given by a specified time (30 to 90 days) prior to the contract expiration should be carefully reviewed. You will need to keep a careful record of any such deadlines in the contracts you supervise to never miss a termination notice deadline.

Limitations on Warranty: This somewhat archaic term derives from a party's promise (or "warrant") that a product would perform in a specified way or fulfill a particular purpose. A warranty on a dishwasher should describe that it will, in fact, clean your dishes. What should be looked for in the software warranty context depends on the nature of the software being procured. For custom software (i.e., software being specifically developed for your organization rather than organizations generally or a specific industry as a whole), you will want the specifications for the development carefully identified in the contract and associated warranties as to key custom functionality. For commercial software (i.e., software already developed for commercial use or for a particular industry – sometimes called "out-of-the-box" software in an outdated referenced to the days when software came in a box), you should be looking for a warranty that the software performs materially in accord with its established specifications and/or documentation (and, for those who want the utmost due diligence, the specifications/documentation should be reviewed). Note that in the context of a most commercial software agreements (which are generally non-consumer in nature), it is typical for the vendor to disclaim statutory and/or implied warranties including "merchantability" and "fitness for a particular purpose"—warranties that typically automatically attach to consumer (and some commercial) transactions. The Entire Agreement clause (discussed above) makes all those sales and marketing statements about how well the software works effectively disappear.

This is where you ask the question, “What happens if it does not work?” Discuss and evaluate the vendor’s support and maintenance process and capabilities. Have the software rep describe the process if a variety of things go wrong. Flaw in the design, needs a “workaround,” loss of access to the software, failure to perform as advertised, etc. Due diligence in this respect is important.

Scope of Work: These provisions are often attached as an exhibit to the “standard contract” and allow the parties a chance to specify the particulars of performance, typically implementation/training and similar services ancillary to the software. This area is where the customer has more leverage to extract specific performance goals. When it comes to implementing software, the implement scope document is one of the most important aspects, as it should clearly describe HOW the software will be setup, configured and similar considerations to meet the customer’s needs. For smaller technology products this may not be as important… but more for complex systems that are integrated with other platforms, the implementation design/scope documentation is a critical component. If something goes wrong, that’s the first place one would look to determine the responsibility for correcting it.

Payment: To the extent possible, shop for comparable prices from comparable vendors. An RFP process may yield some valuable data that might be helpful as you negotiate a price. Another component is timing. Vendors will often “front-load” a software contract to extract a larger portion of the contract price to be paid in the beginning. You retain more leverage if you can make smaller payments over time. For example, in construction contracts, it is not unusual to “retain” a portion of the money owed until the job is satisfactorily completed.

Also consider whether there is a limit on cost escalations. Does the contract price increase based on the number of users/communities, annual increases or both? It’s preferable that the price be locked in for the duration of the agreement, with a maximum escalation for renewals.

Determine if the contract is set up as an overall license, or does it provide licenses based on users or some other metric. Which licensing model (i.e., one-time licensing plus annual support, or based on the number of users, communities, revenue, etc.) depends on the vendor and your specific needs (e.g., if offered, higher up-front cost with lower “downstream” payments versus a more even fee over the contract term). Consider licenses that are based on a role versus individual users; this allows for licenses to be transferred easily upon personnel changes.

Disputes: Although most business disputes result in settlements, there is consensus that mediation as a pre-cursor to litigation is a potentially less expensive route for all concerned. Commercial contracts often specify that all disputes be resolved in the vendor’s hometown (where its expenses will be the lowest).

Insurance: Many contracts contain provisions requiring your company to add the contractor as an “additional insured” to your company’s policy. Sometimes, the proposed

contract will specify a benefit amount (like $1 million). The benefit amount is only part of the story. Each policy carries a “deductible” amount requiring the insured (your company) to pay the first dollars applied to a potential claim. The amount of the deductible, the policy limit and other factors all influence the price of the insurance premiums paid by your company.

Although your employer certainly carries a variety of insurance policies, it is unlikely that you are aware of the current premiums paid, policy limits or deductibles. Find out who has this knowledge and contact them. Explain what the contracts “additional insured’s” language is going to cost you. Sometimes, it is minimal, and a one-page policy amendment called a “rider” is all that is necessary. Sometimes, it will be a deal-breaker that is presenting so much cost/risk that the proposed contractual relationship is not worth that price.

Never agree to an insurance clause unless you have determined the cost.

In addition, you should consider whether the software service provider has the insurance coverage required by your management agreement and that the contract contains the language to document this requirement.

Intellectual Property Indemnification: The agreement should contain a representation by the vendor that the software and/or services do not violate the intellectual property rights of any third party (e.g., patent or trademark rights) and/or an undertaking to defend and indemnify you without limit in the event of any claims or damages to the contrary.

Data Security Provisions: When contracting with a software provider that handles personally identifiable information, personal health information, bank routing and account information, credit card information, company proprietary information or other data considered sensitive, you will want to consider data security provisions for inclusion in the contract. NAA has retained legal counsel to provide you with key information that you and your counsel should be aware of concerning the patchwork of data privacy legislation affecting the rental housing industry. The legislation includes the California Consumer Privacy Act (CCPA), the European Union General Data Protection Regulation (GDPR) and mounting pressure for a national standard. Links to guidance:

https://www.naahq.org/system/files/issues/member-resources/ccpa_update_what_has_changed_and_what_remains_the_same.pdf

https://www.naahq.org/system/files/issues/member-resources/naa_california_consumer_privacy_act_memorandum.pdf

https://www.naahq.org/system/files/issues/member-resources/state-omnibus-privacy-law-comparison-sheet.pdf

https://www.naahq.org/system/files/issues/member-resources/naa_tcpa_compliance_101_memorandum.pdf

If the software gathers, uses or holds private information of residents, prospective residents or associates, including such information as credit card numbers, bank account numbers, social security numbers, birthdates and so forth, than the contract must contain provisions that require the provider to take measures relating to security.

This should include applicable encryption, record retention, controlled access, secure destruction of data when no longer needed and an obligation to comply with relevant privacy and data security laws. The contract should specifically address the provider's responsibility in the event any such data is wrongly accessed, lost or misused.

Data Ownership: The contract should specify that you own the data and your right to obtain and/or transfer data to a destination of choosing. This is crucial should a provider increase pricing terms, go out of business or suffer a breach. This will enable you to move the data to another location.

Sarbanes Oxley (SOX) Considerations: Should your software contract involve the use or management of financial data, and you are a SOX-obligated entity, the software contract may have SOX implications. Please consult with your accounting professionals to ensure the appropriate language is included. At a minimum, you should verify that the software or service provider has appropriate internal controls, including a SSAE18 audits of those controls.

Software as a Service (SaaS) Escrow: Customers ask for SaaS escrows because they worry that the vendor will go out of business and mission critical SaaS will dissolve. The parties solve this problem by giving the software running SaaS app to an escrow agent, who will release it to the customer if the vendor goes out of business. Then the customer can run the SaaS offering. Keep in mind that the SaaS won't run without a platform from third parties, which the customer can't recreate. Do your homework, unless the SaaS is easy to install and operate, the escrow may do you no good.

Service Level and Compatibility: There are many service level and compatibility requirements to consider; following are some key elements but certainly not an all-inclusive list:

- If the solution is a remote web-based solution, the expected minimum standard is 99 percent uptime. Also, consider liquidated damages ("penalties" in layman's terms) for failure to meet this requirement.
- Ensure the contract addresses compatibility with certain browser versions, Adobe versions, and Microsoft Office versions. Consideration should be given to the provider's continued interoperability with new versions or releases but be mindful that your vendor does not control the development path of third-party software providers.
- The contract should identify the vendor's responsibility to maintain or repair the software.
- Ensure the contract specifies the support and training obligations of the vendor and any associated costs. Don't pay for the maintenance, support and training before you need it; wait until he customization of the platform is complete.
- Many companies focus on reducing the sign-on challenges and hence desire to partner with providers that will support single sign-on standards such as Secure Assertion Markup Language (SAML), Active Directory Federation Services (ADFA) or Windows Federation (WS-Fed).

Acknowledgments

Baker McKenzie

Karen Hollinger, Senior Vice President, Strategic Initiatives, Avalon Bay Communities

Womble Bond Dickinson

About NAA

The National Apartment Association (NAA) serves as the leading voice and preeminent resource through advocacy, education and collaboration on behalf of the rental housing industry. As a federation of more than 150 affiliates, NAA encompasses over 82,000 members representing more than 10 million apartment homes globally. NAA believes that rental housing is a valuable partner in every community that emphasizes integrity, accountability, collaboration, community responsibility, inclusivity and innovation. NAA thanks its strategic partners Maintenance Supply Headquarters and Yardi. To learn more, visit www.naahq.org.