

How to Make Software Contract Negotiations Work for Your Business

Jane B. Disbrow

To engage in successful software contract negotiations, businesses must rigorously assess their needs and requirements, understand their providers' strengths and liabilities, and anticipate change.

ANALYSIS

Software contracts are often written to favor application providers and phrased in vague, high-level language that can make companies vulnerable to additional fees or fewer usage rights than customers anticipate. Introducing additional complications, change is the only constant in today's IT environment. Mergers and acquisitions are the norm for companies and their service providers; business models, such as the progressive migration toward outsourcing and globalization, are continually evolving. Consequently, companies must approach contract negotiations as a strategic, iterative element of their business life cycles.

Successful software contract negotiations cannot take place unless the customer:

- Undertakes a rigorous assessment of the company's needs and requirements
- Understands the application provider's strengths and liabilities
- Anticipates changes that are predictable (for example, changes in market demands) and unpredictable (such as natural disasters or mergers and acquisitions)

Gartner clients have many questions about contract negotiations. This Spotlight offers strategies for helping businesses turn the negotiations process into a way to achieve their business objectives.

Business Needs and Requirements

To ensure that a contract satisfactorily addresses needs and requirements, negotiations cannot exist in a vacuum. Companies should designate an "asset manager" from the IT ranks who can identify and enlist a cross-functional, ad hoc negotiation team at the first point of interaction with any application provider. In "How to Create a 'Virtual' Team for Contract Negotiations," Frank DeSalvo provides a road map for establishing such a team. This group should work with all major vendors and contract categories to establish the company's needs and requirements and ensure they will be met. Identifying these individuals can be difficult because their roles don't always directly correspond to conventional job descriptions.

Although the negotiating team must be a "virtual" entity whose composition depends on how the application fits into the corporate structure, the corporate legal counsel is one member who is likely to remain a constant. To meet team requirements and avoid negotiation process delays, develop a list of terminology for advance review by counsel. Once complete, this "checklist" should be reviewed by team members throughout the negotiation. In "Build a Software Contract Checklist," Jane Disbrow offers guidance on this practice and provides advice on some of the more-critical terms in software licensing.

Taking Stock

Ideally, a technology provider should be a partner that provides tools and services to help drive the company's business performance. However, to reach the partnership stage, you must be confident that the provider has the goods for the relationship that you seek. To ensure this, you must establish clear and measurable deliverables to which provider should adhere. If it can't, it might not be the best choice for your business. As Jack Heine points out in "Develop a Framework for Standardizing the Contract Process," the quality of the final "product" hinges on your provider's ability to deliver in three key areas:

- *Components*: The basic materials required to deliver the end product, such as hardware and support labor

- *Processes*: The tasks required to deliver the required product and service
- *Relationships*: The ability to manage subcontracting tasks

After doing the necessary homework, a company might decide that a new, marginal or small application provider supplies the most-appropriate technology for its requirements. This presents the nagging question, "What happens if this provider becomes unable to abide by our service agreement, or worse, goes out of business?" As "insurance" against such a conundrum, customers are increasingly investigating software escrow agreements, wherein they have their software's source code stored, either through the provider or a third party. However, unless such clauses in standard license agreements are explicitly worded, customers should not consider this a guarantee of actually obtaining source code. In "Be Aware of Contract Issues When Negotiating Software Escrows," Jane Disbrow and Alvin Park explain the different arrangements available and the terms and conditions you should include in such clauses.

Anticipate Change

Escrow agreements provide a mechanism for counterbalancing one future concern for licensees, but this addresses just a microcosm of the forward-thinking analysis that companies must conduct when negotiating contracts. In "How to Make Enterprise Licensing Work for You," Frank DeSalvo explains how licensees should define the scope of the term "enterprise" in any contract. Trends on the rise in the business climate, such as mergers and acquisitions and outsourcing, mandate that the negotiating team must consider how future developments and issues, both internal and external to the company, could affect its structure, software usage requirements and terms of use.

Other research in this Spotlight addresses specific scenarios. In "Software Contract Considerations for Mergers, Acquisitions and Divestitures," Alvin Park describes the implications of some of the more high-profile "changes" that companies encounter, which often come without warning — even for senior IT staff members. Many software contracts require that businesses pay additional fees when combining assets. When entering or renewing any software contract, companies should review and amend merger, acquisition and divestiture clauses (at no additional fee) and the right to transfer licenses. Carefully constructed clauses can provide value in mergers and acquisitions.

Outsourcing is another business decision that can add complications to software licensing. Software contracts often include restrictions that can significantly increase cost and complexity. Some applications, for example, can only be run on the client's site or on the vendor's hardware, even down to a specific serial number. Many application providers use "business metrics" (such as the number of users) to determine license costs, rather than hardware capacity. License agreements should explicitly state that your company's business, rather than the outsourcer's business, will be the gauge by which you'll be charged. In "Protect Yourself From Software Complications When Outsourcing," Richard Matlus, Frank DeSalvo and Jane Disbrow explain how to negotiate favorable terms.

In "How IBM, Microsoft and Oracle Address Disaster Recovery and High Availability Server Software Licensing," Alvin Park, Jane Disbrow and Donna Scott address another pressing concern when negotiating software contracts: the leading providers' policies and prices for software that runs on high-availability and disaster-recovery servers. For instance, for warm failover — the most popular type of disaster recovery architecture — prices and policy vary significantly. The arrangements range from no cost from IBM if the warm DBMS is inactive, or a single processor charge if the DBMS is active; Microsoft's 25 percent license fee (with software assurance); and Oracle's full-license charge.

REGIONAL HEADQUARTERS

Corporate Headquarters
56 Top Gallant Road
Stamford, CT 06902-7700
U.S.A.
+1 203 964 0096

European Headquarters
Tamesis
The Glanty
Egham
Surrey, TW20 9AW
UNITED KINGDOM
+44 1784 431611

Asia/Pacific Headquarters
Level 7, 40 Miller Street
North Sydney
New South Wales 2060
AUSTRALIA
+61 2 9459 4600

Latin America Headquarters
Av. das Nações Unidas 12.551
9 andar—WTC
04578-903 São Paulo SP
BRAZIL
+55 11 3443 1509