IADC COMMITTEE NEWSLETTER

CYBER SECURITY, DATA PRIVACY AND TECHNOLOGY

February 2021

IN THIS ISSUE

Anna Cook and Robert Graham of Bristows LLP discuss what happens when the legal and commercial mechanisms in a complex IT contract are not used by the parties.

If you Want to Stay Friends with your IT Supplier, Use the Contract!

ABOUT THE AUTHORS

Anna Cook and Robert Graham are litigators at Bristows LLP, specialising in technology law, IT projects and the use and exploitation of software and IP. Each of them has extensive experience of troubled IT projects and the litigation that follows as a result. Anna can be reached at Anna.Cook@Bristows.com. Robert can be reached at Robert.Graham@Bristows.com.

ABOUT THE COMMITTEE

Corporations and law firms around the world are constantly dealing with cybersecurity, data privacy and other important technology issues, both in business and, where available, in litigation discovery. Burgeoning technologies are placing new and increasing demands on in house and outside lawyers and their clients. All are being challenged to meet new and strict data privacy and security guidelines and the consequences for failing to meet these requirements can be devastating. The Cyber Security, Data Privacy and Technology Committee will address the differing substantive laws globally in these areas, and be of interest to many other committees whose members and activities are impacted by technology. Learn more about the Committee at www.iadclaw.org. To contribute a newsletter article, contact:

Avanti D. Bakane
Vice Chair of Publications
Gordon & Rees, LLP
abakane@grsm.com

*The **International Association of Defense Counsel** serves a distinguished, invitation-only membership of corporate and insurance defense lawyers. The IADC dedicates itself to enhancing the development of skills, professionalism and camaraderie in the practice of law in order to serve and benefit the civil justice system, the legal profession, society and our members.*

w: www.iadclaw.org   p: 312.368.1494   e: mmaisel@iadclaw.org

When it comes to implementing IT systems, most businesses use carefully negotiated, nuanced and lengthy contracts. One of the core problems, however, is that the nature of the projects means that the parties often avoid using the contractual mechanisms and rights available to them. Lots of careful work to record the balance of risks and to maintain the commercial tensions and incentives can come to nought.

In this article we review:

(a) the characteristics of these contracts;
(b) why things go wrong; and
(c) the impact of not using the contractual mechanisms or rights in the contracts.

Defining the Deliverables

In the context of a complex IT project, such as an outsourcing or business transformation exercise, the customer often needs help to define its future "end state". Although the customer might know the outcomes it wants to achieve ("*We want a better billing system with fewer manual inputs*"), it will not necessarily know what is possible or how far its own internal processes deviate from the norm. Therefore it has become common for the supplier, as the expert, to analyse the customer's requirements and to explain what is possible. In a complex project, this goes much further than a tender exercise. Often the customer needs substantial consultancy work to analyse its own processes, to identify inefficiencies (or deviations from "best practice"), to identify software requirements and for the supplier to suggest and design software solutions.

This creates a risk: the customer is in its supplier's hands. It is a particular danger if the business has no coherent vision for what it wants – and the risk is multiplied for complex businesses. A "vision" articulated at the top of the business is often difficult to communicate internally. In particular, individuals often struggle to imagine how software will work and to articulate how it might deliver what they actually want.

In this situation, regardless of any contract, the commercial starting positions are unequal because the customer is so highly dependent on the supplier. The customer also needs to take care to ensure that the definition of success (and contractual performance) are aligned with its objectives, even though it may not be able to specify its requirements.

The other issue, frequently underestimated, is that IT projects can create enormous business change, and that such change needs to be absorbed and accommodated by employees and other stakeholders. The commitments required from a customer can be very substantial.

Specific Risks

Use of Packaged Products

Businesses often adopt standard packaged or cloud based software (i.e. an accounting package) and make their own processes fit the software. In other words, the business decides to modify its own ways of working to match the processes demanded by the packaged software, rather than to incur the costs of building something bespoke. This is also recognised as a good way for the customer to obtain the benefit of well-established processes and common standards.

In larger and more complex projects, suppliers are often asked to use or to integrate standard packaged software within a larger system. This involves the supplier in three activities: (i) to identify the customer's requirements, (ii) to understand the standard package(s) and (iii) to design how to bridge any gaps between the standard packages and the customer's requirements.

The benefit of adopting a standard package in a large project is that it gives the customer an external benchmark for some requirements and, potentially, a minimum standard. However, this approach does not necessarily simplify the project. Although the packaged product(s) might be a good match for the customer's basic requirements, the work required to build the rest (i.e. to bridge the gaps) can be highly complex and time-consuming.

From the supplier's point of view, the use of packaged software creates certain risks. First, that the customer might not understand the real complexity of the project. Second, there may be delays caused by issues outside of the supplier's control, affecting its own ability to deliver. Third, the supplier is highly dependent on the reliability of the information and documentation relating to those third party products. Fourth, if there are changes to the scope or the requirements, the implications can be extremely difficult for the supplier to assess and to understand.

The use of standard packages can also cause the customer real issues. In the event of delays or problems, it may have to ascribe blame amongst several suppliers each of which may have a different contractual relationship with them. It is also extremely difficult for the customer to evaluate progress at any time and to work out how close it is to the "finish line". Finally, if there are (or have been) any changes, including changes to the customer's own requirements, it can be complex for the customer to understand the real costs and impacts.

Therefore, even though the parties may have agreed a formal regime to manage and escalate problems, to deal with change or to ensure that delays are quickly mitigated, all of these mechanisms depend on good information about the cause of the problems. In the absence of information, the parties need to decide whether to trust each other.

If the parties trust each other, it becomes much easier to make an informal agreement about how to tackle the immediate issue and therefore to avoid the contractual mechanisms and other formalities. However, trust can quickly dissipate if there are repeated problems. A common feature of these projects is that the initial trust leaves a legacy of undocumented change and undocumented "side deals" about delay. As these accumulate, they can amount to a substantial departure from the written contract and cause significant evidential and legal issues in the event that a formal dispute arises.

Identifying Requirements and Contracting

Often the customer needs help from the supplier to identify its requirements and to advise on what might be possible. At some point, in order to create an enforceable bargain, the supplier and the customer have to describe the deliverables and to define what the supplier is doing.

In practice it is very difficult and time consuming to create an enforceable descriptive document, even if the customer has expert assistance. Any technical specification is likely to contain a very large number of unwritten assumptions and implied terms that would not occur to the people writing it. In other words, even if the customer sets out to create a document that is fully capable of being enforced, the customer still bears the risk that a fully written specification is imprecise and will not deliver what it wants.

In order to identify the customer's requirements, the most common approach is for the supplier (or sometimes an independent consultant) to lead workshops with the customer and to record the requirements through a process of listening and feedback. If this is done well, the expectation is that the parties avoid miscommunication and achieve a strong mutual understanding of each other's positions. This approach should ensure that the parties are communicating effectively and that the supplier has a complete understanding of complexity and scope. However, this is a time-consuming process that demands a major commitment from the customer's internal subject-matter experts.
Once this process is done, the supplier who has led the process is in a strong bargaining position; in order for the customer to have the benefit of the costs sunk into the process of achieving mutual understanding, it needs to retain the supplier's delivery team. Therefore it is relatively uncommon for the customer to do the requirements capture exercise and the workshops before it has locked the supplier into the full contract.

In order to achieve this "lock", the customer needs to create a contract which is flexible enough to address its "known unknowns". This creates tensions at the contracting stage. If scope and complexity are variable elements, the parties need to decide which of them will bear the risk that the project will cost more and/or take more time than budgeted. Again, it is important to recognise the strength of the supplier's underlying commercial position because, as soon as the customer sinks costs into the project, it is committed, especially in circumstances where the customer has a pressing business need for the project to be delivered.

Customers who are in a strong bargaining position will often try to create a fixed price contract for a principled, broad scope of work. The approach often involves identifying "outputs" (i.e. *"All letters received by post shall be scanned and stored in the correct matter file in the database"*) and "user stories" (i.e. *"a partner shall be able to access and review all incoming mail in their allocated matter files"*). In most of these contracts, the implicit requirements and the details about how the functions will be performed are also within the fixed price and the supplier's scope of work. Conversely, suppliers who are in a strong position will incorporate fixed and variable price assumptions, for example by specifying the amount of time or effort they will expend on specific deliverables.

In truth, it is often in both parties' interests for the contract to be reasonably balanced so that the project is approached with suitable realism and so that the trade-offs and incentives are well understood. It is usually in both parties' interests to complete the project as quickly as possible and with minimal work. That may mean the customer compromising some of its requirements or the supplier bearing some additional cost.

When it comes to disputes about scope, large projects can be especially vulnerable to changes in personnel and the loss of that shared mutual understanding. This means that the supplier and the customer can easily become embroiled in disputes about implicit requirements and whether they are within scope or whether they should be the subject of the change mechanism (more time and more cost for the customer). Again it is quite common for the parties to start off by trying to resolve these disputes on the basis of goodwill and reciprocity. This often happens if the parties are under time pressure or if there have been earlier delays or breaches. Neither party wants full confrontation.

However, the parties' interests are different. When it comes to disputes about scope, the customer is likely to be seriously interested in ensuring that the disputed requirements are fully documented and, for that, it needs to engage formally with the supplier. If the supplier hopes to resolve the issue informally, it has no real interest in such formality. The impact is that these change/scope disputes can take a long time to resolve.

Problems and Breaches

A very high proportion of IT projects get into difficulties. Usually the first problem is that of delay because these projects often have large numbers of interdependencies. For example, where the customer's subject matter experts are relied on to inform the supplier about the requirements or to test and give feedback about the deliverables, they may not easily be able to accommodate a changed or extended timetable alongside their day job. This means that disputes about scope or change become doubly dangerous because a delay in resolving them could have further knock-on impacts.

There is also a very close relationship between time and the quality of the deliverables. If the supplier is obliged to deliver software before being ready to do so, it may need to cut corners. This may mean, for example, shortening the cycles for testing (which tends to lead to more bugs and errors) or delivering software that is incomplete.

Once a project starts to suffer from delay, the supplier has strong incentives to control and limit the scope of work, to manage the customer's expectations and, in cases of serious delay, to prioritise essential functions to be delivered ahead of the rest. Although customers usually want the project to be finished as quickly as possible, they often have incentives to grant the supplier more time in order to get better quality deliverables. Therefore, once a delay comes to light it can often cause an urgent need to formalise the status of the project. This can coincide with a complete breakdown of trust. This means any undocumented changes and assurances are put under the microscope. The experience of delay means that the parties have to confront the gaps and uncertainties in the original contract.

<u>What is the Impact of not Using the Specific Rights and Mechanisms in the Contract?</u>

There is no contract so good that it can guarantee the supplier's IT deliverables will completely satisfy the customer. Indeed, for the reasons outlined above, most IT contracts contain a large number of "known unknowns". The best that can be achieved is for the parties to establish strong frameworks to deal with uncertainties as they arise. However, it is also vital for the parties to use those frameworks and to take conscious steps to overcome any natural reluctance to engage with problems.

The parties also need contract managers who are able to spot problems and to act on them. They need to be prepared to be confrontational. Confrontation is difficult in circumstances where neither party is certain of its position. However, in most cases, it is better or preferable for the parties to avoid inertia, make a decision and document it, rather than to allow scope and delay problems to accumulate.

The failure to manage these projects can sometimes lead to a breakdown in the relationship. If this results in litigation, the disputes can be highly complex, leaving the lawyers to piece together the specification (i.e. to define what should have been delivered and to explain what is missing) in the absence of a clearly enforceable list of requirements. Often though, the customer ends up with only part of what it wants, a "minimum viable product", at more cost and later than it expected. Even in this scenario, it is very difficult to define what has not been delivered.

The parties may also have claims against each other for their contributions to delay. For example, the customer might claim the costs to maintain its legacy systems and the supplier might claim the costs of keeping its project team engaged during a period of delay. This type of claim involves a very detailed examination of the causes of delay and the way in which problems were handled as they arose. If the parties have not used the contract management mechanisms, there are likely to be claims that the timetable was varied by agreement or that there has been a waiver. Many of these issues could be avoided, or at least minimised, if the parties operated the contractual mechanisms which address responsibility for delay timeously (such as seeking a time extension in the event of a default), rather than attempting to do so retrospectively when trust has dissipated and a dispute has arisen. Large projects tend to have complicated project plans, with significant dependencies on the customer and potentially third parties. This means that the questions of causation are highly complex and inherently open to obfuscation by a party seeking to avoid responsibility.

Finally, the customer may have claims that the quality of the software is poor and, for example, it does not meet minimum expectations about reliability or performance. Usually a failure to meet contractual standards can be independently tested by an expert. However, if there have been variations to the contract, this can complicate the analysis, especially if parts of the system have been accepted or are in live use. Quality problems become more likely if the project has been affected by other issues.

In summary, where a complex solution is being developed, it is very difficult to simply rely on the contract to impose commercial risk on the other party. Unless the customer is really prepared to walk away from an ongoing project, it will be vulnerable to claims by the supplier for more time and more money. Unless the supplier has accurately recorded the scope and the scale of effort to deliver it, the supplier is also at risk of losing money. Therefore, it is essential for the parties to be clear-eyed about their mutual incentives and to focus on constructive problem solving and rapid decision making. Most contracts provide the mechanisms for this, but it is up to the parties to use them.

Past Committee Newsletters

Visit the Committee's newsletter archive online at www.iadclaw.org to read other articles published by the Committee. Prior articles include:

MARCH 2018
First Class Action Data Breach in the UK – Employer Found Vicariously Liable for Rogue Employee's Actions
Elena Jelmini Cellerini and Vikram Khurana

NOVEMBER 2017
Blockchain Unchained: One Lawyer's Quest to Figure out what the Hell Everyone is Talking About
Kendall Harrison

AUGUST 2017
Def Con Hacker Conference: An Accidental Tourists Observations
Elizabeth S. Fitch

MAY 2017
Doe v. Backpage.com and its Aftermath: Continued Uncertainty and New Litigation in the Wake of the Supreme Court's Denial of Certiorari
David Patrón

APRIL 2017
Incorporating Technology into the Management of the Work Processes at the Firm
Donna L. Burden, Elizabeth S. Fitch and Park L. Priest

FEBRUARY 2017
"The First Thing We Do, Let's Kill All The Lawyers"
Elizabeth S. Fitch and Elizabeth Haecker Ryan

DECEMBER 2016
A Primer for Understanding Blockchain
Doug Vaughn and Anna Outzen

AUGUST 2016
The Attorney-Client Relationship in the Electronic Age
Elizabeth S. Fitch and Theodore M. Schaer

MAY 2016
Understanding the Defend Trade Secrets Act of 2016: "We're Not in State Court Anymore"
Peter J. Pizzi and Christopher J. Borchert

DECEMBER 2015
Cyber Armageddon: Survival or Annihilation?
Theodore M. Schaer and Elizabeth S. Fitch