

scripted |

Volume 17, Issue 1, January 2020

Book review: *IT Contracts and Dispute Management: A Practitioner's Guide to the Project Lifecycle*

Steven Baker, Lawrence Akka QC, Rachel Glass
Cheltenham, United Kingdom: Edward Elgar, 2018. 448 pages.
ISBN 9781784710118. £145.

*Reviewed by Cliona Kelly**



© 2020 Cliona Kelly

Licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license

DOI: 10.2966/scrip.170120.166

* Assistant Professor, Sutherland School of Law, University College Dublin, Dublin, Ireland Email: cliona.kelly@ucd.ie.

There is often a perceived gap between the principles of contract law as stated in textbooks and the business realities of how contracts are entered into and projects managed and delivered. The authors of *IT Contracts and Dispute Management: a Practitioner's Guide to the Project Lifecycle* effectively and impressively bridge that gap. It contains a detailed exposition of how the law – or, more accurately, the law in England and Wales – applies in reality to the management of a technology project, from the very early stages of its development, right through to issues such as litigation and dispute management. The book's core strengths are its ability to weave legal principles into a practical context, and its identification of steps which can be taken in the management of an IT project to avoid disputes. The preface to this text explains that it was “conceived through painful professional experience”, but it should go far in helping others learn from this experience.

The first two parts of this text takes us through the planning stages of a project through to maintaining and delivering the project. The authors outline practical ways in which practitioners can act in the early stages of a project to avoid disputes further down the line. They identify key risks (e.g. commencing performance before a contract has been finalised) and ways to reduce the possibility of dispute where taking such a risk is unavoidable (e.g. by entering into binding agreements relating to discrete aspects of the project to enable preliminary work to progress). The authors show how, at the drafting stage, contracts need to be adapted to take into account the features of any given project and the contemplated style of project management. Boilerplate terms tend to be drafted with more traditional styles of contracts in mind, and this may not always

suit the flexibility and levels of cooperation that are needed in IT contracts, particularly agile projects.¹

Before a dispute arises there are often operational disputes and delays over the course of an agreement – minor breaches, variations and concessions that are often not fully acknowledged or documented. The authors emphasise the need for good “housekeeping” of these changes, again against the context of contract law principles such as estoppel by convention, the rules on variations, and the rules on giving of notices. In relation to the rules on variations, it should be noted that the Supreme Court has since ruled that an “anti-oral variation clause”, i.e. clause in a contract which requires modifications to that contract to be in writing, invalidates a subsequent oral agreement to vary the contract.² Again, the difficulty that arises is that technology contracts “often require significant cooperation between supplier and customer”³, relying on repeated development iterations rather than traditional detailed specifications, and contract law does not always clearly allow for this. The authors do an excellent job of bridging the gap between contract law doctrine and the practical realities of technology projects, for example by advocating the use of change control processes to set out project requirements and to keep to the terms of the contract, and suggesting that the parties have access to the contractual documents, including any amendments, for reference. The legal framework for features that might be found in long term contracts – such as benchmarking or service credits regimes – is also set out. There are also clear and accessible discussions on regularly used but often misunderstood concepts such as “reasonable” and “best” endeavours, and on the (seemingly ever expanding) use of indemnities.

¹ See the Manifesto for Agile Software Development <https://agilemanifesto.org/> and more information at <https://www.agilealliance.org/>.

² *MWB Business Exchange Centres Ltd v Rock Advertising Ltd* [2018] UKSC 24.

³ p. 84.

The third part of the book dramatically (but accurately) refers to “Project Rescue” (Chapter 11) and “Project Resuscitation” (Chapter 14), and examines the most common approaches to resolving more serious disputes that arise mid-project, such as “de-scoping” problematic areas.⁴ The authors emphasise the need to ensure that any such resolution aligns with the underlying contract, as any disagreement later on could result in extensive and complex disputes. Methods of interim dispute resolution such as escalation clauses and dispute boards are also discussed.

Some contracts might be beyond saving, in which case the ability to exit the contract is key. The book contains a lengthy and detailed discussion of the circumstances in which termination is possible, including a very useful and clear exposition of the differences between contractual termination rights (e.g. for a “material” breach), and termination at common law (e.g. for a repudiatory breach), as well as the interaction between the two. Again, the authors do an excellent job of applying general contract law principles in the context of an IT project, and include an extensive discussion of tactical and strategic considerations when considering settlement and resolution options. In relation to the issue of damages, long-running disputes in the technology sector can pose particular challenges because of the rapid rate of change which “can cause the ‘contracted-for’ solution to be rapidly overtaken in the course of implementation”,⁵ causing difficulties in quantifying lost benefits.⁶

⁴ The authors (at p. 141) describe “descoping” as “carving out tranches of work by stage, by deliverable or by milestone, and agreeing that they are abandoned completely or effectively postponed and incorporated (in full, or in an abbreviated form) at a later stage of the project.” Later delivery of difficult sections of the project “provides some ‘breathing space’ for the parties”.

⁵ p. 327.

⁶ See for example *BskyB v EDS* [2010] EWHC 86 (TCC).

The final section of the book outlines various means and forums for resolving disputes, including the civil court system, arbitration, mediation and expert determination, with guidance on the rules on disclosure and document preservation and the role of expert witnesses.

This is an excellent resource, which benefits from the professional experience and personal insight of all three authors. It clearly connects principles of contract law to the practical realities of the life cycle of a technology project, and offers strategic and coherent advice on not only how to avoid disputes, but how to manage them if they arise.