Wigley & Company

BARRISTERS and SOLICITORS

Intellectual Property in IT Contracts

July 2006

1. Introduction

Increasingly we're finding that one of the most time consuming and difficult areas in IT contracts is ownership of intellectual property ("IP"). Off-the-rack clauses and precedents are often inadequate and need to be customised to suit the circumstances. In this paper we will focus on the situation where software is being developed for a customer, rather than the customer purchasing off-the-rack licensed software such as Microsoft products.

Historically, there hasn't been particularly close attention paid to the IP clause in IT contracts. But recent developments show this should be an area of focus. These developments include: the value of the IP; benefit for the customer; and legal cases which illustrate risk. Customers (particularly public sector agencies) tend to start from the assumption that they should own the IP. But that's not the only, nor necessarily the best, option for the customer. And vendors can have problems if documentation is inadequate.

2. How vendors protect themselves

If a vendor wants to retain the rights in the software, they'll generally license it. The main method of protection in that context is to give to the customer a version of the software that, in theory, can't be modified. In fact that version may well be able to be "reversed engineered". But that's difficult, and contracts generally contain clauses prohibiting this. So, in practice, the vendor will retain the source code as a method of protecting itself. If a customer is taking title to the software, it will want to have the source code.

3. Licences – risks for customers

Where a customer takes the software under licence, it faces a risk if the vendor goes under or fails to perform. One way it can protect itself is to have a copy of the source code lodged with a trusted third party, by way of a software escrow arrangement. The source code is then made available to the customer when the vendor defaults. This is an acceptable approach provided the arrangement is set up well. Often it is not. See the benefits and pitfalls in our paper "Software Escrow Agreements: a Business Continuity Strategy".[1]

4. Who should own the IP developed for a customer?

The simple answer often arrived at is that the customer should own it. It pays for the software development, so why shouldn't it own the software?

That in fact reflects the usual default position under the Copyright Act, both under:

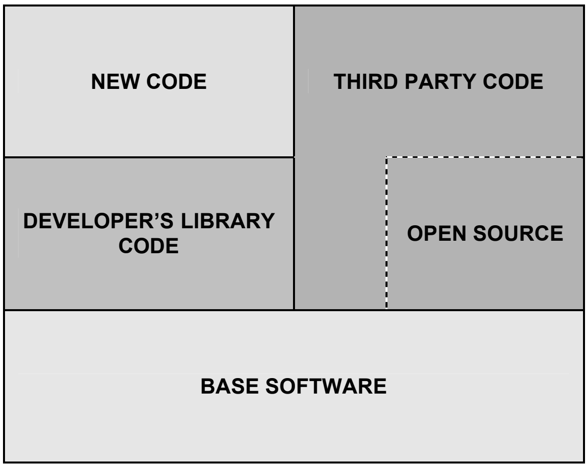
- section 21 (which deals with first ownership of copyright); and
- where the customer is a public sector agency, section 26 (which deals with Crown copyright).[2]

Both sections 21 and 26 are subject to any contrary agreement. They can be overridden by contract and often are.

Very often, getting ownership of the IP is of much less value to the customer than appears at first sight. Usually a customer is not just having new code developed for it. That code is generally built upon other building blocks, as the diagram below shows. For many of the components in the final product, the customer can only get a licence.

[1] http://www.wigleylaw.com/SoftwareEscrow Agreements.html

[2] Note that there are some definitional issues around what is included within "Crown" for this purpose (see section 2 of the Act, which, for example, precludes the application of section 26 to Crown Entities and SOEs). Also, some IP being developed for the Crown may not fit automatically within section 26.



It's typical for a new software product being developed, to be built on an existing base. That base might be owned by the developer, some third party, or even the customer itself.

Frequently, the developer will use its own library code. It will want to use that code for other customers as well. The customer benefits from this of course as it reduces price theoretically. The developer won't want to give ownership of that IP to the customer, nor is it necessarily in the customer's best interests to do so. So it will licence library code to the customer. This is a bit like lawyers re-using precedents and earlier documents for subsequent clients.

Often, the final product will also include third party software which is licensed from elsewhere (eg, Microsoft products).

Sometimes, open source software is included. This is a particular risk area. Those drafting contracts should always ask the technical people whether open source is included in the software as this presents particular risk areas. We should be looking closely at the underlying licence and other arrangements.
Contrary to the diagram, the various software components can in fact be interwoven in a way which is not immediately clearly distinguishable. It all depends and varies according to the circumstances.

As is apparent from the diagram, very often the newly developed code is only a small part of the total product. Getting ownership (and therefore the source code), for that component, is usually of very little value to the customer. Often the customer can do little with it. If it could do anything, it may be an expensive process.

5. What does the customer really want?

What most customers really want is a product that works, at a favourable price. Whether or not they own the IP is really secondary to this as long as that outcome is achieved. Generally, the customer won't want to on-sell or commercialise the IP (unless, say, it is an entity such as a CRI or it's in the IT business).

By owning the IP in the newly developed product, the customer may well be worse off. If the vendor (or some other party which can commercialise the software) is able to sell the IP elsewhere, then:

- the customer should be able to buy the product at a cheaper price; and
- as the product is rolled out to other customers, faults in the software are picked up more quickly, and the product is more likely to be upgraded down the track. In other words, the customer is more likely to end up with a more robust and better product.

6. Risks for the customer

Of course, if the customer takes the licence rather than ownership of the IP, it generally won't get the source code. One way of protecting the customer in this situation is to get the software put into escrow, although there are the pitfalls noted in the article referred to above.

Importantly, whether or not the software is owned or licensed, the customer is normally at just about as much risk either way. If the vendor goes out of business or defaults, it's likely the customer will have a problem anyway.

In terms of solutions, there are a number of options including:

- the vendor owning the IP and the customer taking a licence (perhaps with escrow);
- the customer owning the IP and the vendor taking a licence (so that it can sub-license and commercialise); or
- joint ownership.

Joint ownership is superficially attractive. But it carries risk if the rights of each party are not carefully documented, including rights to use the IP for particular purposes without having to get the other party's consent. The legal position is not clear, but it may well be that joint ownership is as tenants-in-common. So both must agree before anything can be done with the software. To avoid that situation there must be clear agreement to the contrary. Often a different approach is better, such as one party owning and the other licensing. But carefully defined joint ownership is still an option.

7. Copyright Act – the default position

It is particularly unwise to rely on the default position in the Copyright Act, as this can be messy. It's far better to set out the detail in the agreement. For an example of some of the problems in a public sector context see *Advanced Management Systems Ltd v Attorney-General*.[3] This case went to appeal on a different point: *New Zealand Payroll Software Systems Ltd v Advanced Management Systems Ltd*.[4] In a private sector context see *Pacific Software Technology Ltd v Perry Group Ltd*.[5] We deal with that case below as it's such a good example.

8. Public sector issues

The public sector forms a very large part of the IT sector. Public sector agencies have the ability to benefit the New Zealand economy as a whole by facilitating commercialisation of IP. They can also benefit their own interests by enabling commercialisation (which should lead to reduced pries and better products as the software is rolled out elsewhere). There is a firm push in this direction, including by way of MoRST and by way of NZTE/MED initiatives such as ICTX.

It's expected that good guidelines in relation to IT IP will be developed for New Zealand. In the meantime Australia's Commonwealth IT IP guidelines are very helpful.[6]

As those guidelines note:

> In contracts relating to IT-related IP, agencies should not automatically assume that all IP rights must be vested in the Commonwealth, but should consider whether vesting IP in, or granting a licence to, the supplier or contractor might yield savings and a product that in the long term more effectively meets agency objectives.

This contrasts with MoRST guidelines for research projects.[7] They push firmly for IP ownership by public sector agencies. These 2-page rules, which apply to publicly-funded research, not directly to IT projects, are not prescriptive. They do encourage licensing for commercialisation, etc, where appropriate. It is to be hoped that similar guidelines for IT will facilitate a much wider array of options, along the lines of the much more comprehensive Australian guidelines.

9. Third party licences

A particular risk area for customers is the use by a vendor of software from third parties. Unwittingly or deliberately, the vendor may end up using software that it is not entitled to use. In terms of balancing risk between vendor and purchaser, generally this risk is put on the vendor. Often this is done by way of an open ended IP indemnity. These clauses should be reviewed carefully, however. Often they restrict the vendor's responsibility to finding a solution, or the customer getting their money back. High profile cases such as the SCO litigation against IBM highlight the very real risk in this area. This can impact on both public and private sector customers.

10. Practical example – the Perry case

The facts for this case are not unusual. The developer had worked in IT for over 15 years and managed his family owned software company. The client, a dealer of gaming machines, had just employed a new systems manager who had previously worked with the developer and knew him well.

---

[3] *Advanced Management Systems Ltd v Attorney-General* (30 November 2001) HC AK CP 371/00, Anderson J.
[4] *New Zealand Payroll Software Systems Ltd v Advanced Management Systems Ltd* [2003] 3 NZLR 1, Tipping J for the Court.
[5] *Pacific Software Technology Ltd v Perry Group Ltd* [2004] 1 NZLR 164.

[6] http://www.dcita.gov.au/ip/commonwealth_it_ip_guidelines/the_commonwealth_it_ip_guidelines
[7] http://www.morst.govt.nz/?CHANNEL=Intellectual+property&PAGE=Intellectual+property

It wasn't long before the developer learnt of some of the deficiencies in the client's existing database. Here was an obvious opportunity for the developer to secure new business. Sure enough, after a series of informal discussions, and no contract, the developer started working on a new software database. As part of this development the developer utilised its own library code, information gained from the client's employees, and their old database.

However, problems arose when the parties tried to negotiate a joint venture to market the software to other players in the industry. It soon became apparent that the developer and the client were poles apart on the issue of ownership.

The client claimed ownership of database as the client paid for it and had commissioned the developer to do the work. The developer's view was that his company had ownership because it was an independent development project. That is, they had seen an opportunity to create the database and had developed it independently of the client with a view to marketing it to a number of clients.

So what's the legal position?

As we note above, the starting point is New Zealand's Copyright Act. This states that copyright can vest in a computer programme and, by default, the author of a computer programme is its owner. However, if another person has commissioned and paid (or agreed to pay) for the making of the computer programme, that person will be the owner.

So, the issue for the court in Perry was whether the development of the database had been "commissioned" by the client or had been carried out independently by the developer.

The court's efforts to resolve the issue were hampered by what the first judge referred to as a "paucity of documentation". In short, there was no contract and no correspondence that indicated how the parties embarked on the project. This sort of situation happens a lot.

After examining what evidence there was, the High Court decided that the database had been commissioned by the client. It did not matter that the "commissioning" was informal, nothing was in writing, and there was no agreement on a fee or a deadline.

A strong factor in convincing the court was the fact that the database was specific to the client's requirements and incorporated a substantial amount of the client's confidential information. If the software was independently developed to be sold to various parties it would have been more generic and not so client-specific.

So in the High Court, the client got full ownership of the software – to the detriment of the developer. The developer lost ownership of all the components of its own library code that it had included in the database. Any suggestion of an implied licence or even joint ownership was rejected by the court. The court's view was that the client commissioned the whole of the software and so was entitled to complete ownership.

Not surprisingly, the developer decided to appeal. Against the murky factual background, the Court of Appeal was quick to conclude that the Copyright Act default position applied, that Perry Group had commissioned the work, and so they owned the copyright.

However that outcome alone would still leave developer having lost copyright in its library code. The Court of Appeal said this should be treated separately and that it was still owned by the developer. To give practical effect and enable Perry Group to use the software as a whole, the Court implied a term by which the library code was irrevocably licensed by the developer to Perry Group and its assignees. A pragmatic solution to an awkward problem. The Court of Appeal decision is reported at [2004] 1 NZLR 164.

The Court of Appeal reiterated the point that the default position under the Copyright Act:

> …does not diminish the importance of sound industry agreements. On the contrary, developers would be very well advised to protect themselves by agreement, since otherwise the onus will, in practice, be on them to demonstrate the existence and extent of alleged prior rights and the limits of any implied licence. Secondly, it does not follow that the reasoning in this case will necessarily follow in every case: whether or not there is an implied licence (and the terms of it) will necessarily depend on the facts of the given case.

Naturally, decisions like this can make developers nervous. Many projects start off without any documentation and it's often assumed that the developer retains the code. Moreover, from a sales point of view, it's not easy to trot out the legalese when the relationship's good and there's a prospect of securing a client.

However, as Perry shows, the risks of not reaching an agreement on ownership are real. It's one of those cases where an ounce of prevention is worth a pound of cure. Here are a few suggestions:

- **Work it out:** Figure out what your intellectual property is and what it is you need to keep.
- **Talk it out:** Half the battle is won if the parties sit down and talk through their respective interests. The key here is to think through what each party needs. For example, if the end product is very unique, specific to the client and unlikely to be reused by the developer, it may be that the client should get ownership (but with library code rights retained by the developer). On the other hand, if the client is not planning to resell or modify the products themselves it may be that a licence to use will be sufficient.
- **Write it out:** It doesn't need to be the mother of all contracts, but it should be tailored to the circumstances.
- **See it out:** A project can go in all sorts of directions. So it pays to keep an eye on any new developments. If the deliverables have changed, maybe the agreement on ownership needs to be tweaked.

The points above seem straightforward, and often they are. However, if you are in any doubt, get legal advice. The cost of getting it wrong can be high. If you're not clear who should own what, you can end up getting stung at the end.

We welcome your feedback on this article and any enquiries in relation to its contents. This article is intended to provide a summary of the material covered and does not constitute legal advice. We can provide specialist legal advice on the full range of matters contained in this article.

Wigley & Company is a long established specialist law firm. Our focus includes IT, telecommunications, regulatory and competition law, procurement and media/marketing. With broad experience acting for vendors and purchasers, government agencies and corporates, Wigley & Company understands the issues on "both sides of the fence", and helps clients achieve win-win outcomes.

With a strong combination of commercial, legal, technical and strategic skills, Wigley & Company provides genuinely innovative and pragmatic solutions.

Wigley & Company
Barristers & Solicitors
Level 7, 107 Customhouse Quay
P O Box 10842
Wellington
info@wigleylaw.com
Tel: 04 472 3023
Fax. 04 471 1833