SOFTWARE ESCROW AGREEMENTS:
A BUSINESS CONTINUITY STRATEGY

PAPER PRESENTED FOR
NEW ZEALAND COMPUTER SOCIETY
"THE LAW OF IT" SEMINARS

The New Zealand Computer Society Inc.

Wellington and Auckland
March and April 2003

Where software is licensed, one of the strategies to minimise the licensee's risk is to have the software held in escrow with a trusted third party. This really is a business continuity strategy and is not without its difficulties. We address some of the strengths and weaknesses and approaches that can be taken.

Software escrow agreements are not used as much as they should be used. While not a perfect solution they do offer a way of significantly reducing risk. And they provide marketing benefits for local developers.

INDEX

1. The Problem

1.1 Acme Trading has software developed for it by CodeCutters Limited. But then CodeCutters goes under. Acme - as usual when software is licensed – doesn't have the source code. What can Acme do when it needs to fix the software, upgrade it, etc? If Acme can't do this, a critical system fails and they can't deliver multi-million dollar services to their customers. They could be sued for millions for failing to supply their customers.

1.2 Software providers going out of business or failing to support their software is a very real issue. If the software is important or critical to Acme's business, this is a significant business continuity planning issue, especially if Acme can't get replacement software quickly from elsewhere. There is unlikely to be any perfect solution to this problem. The idea is to consider this problem from a strategic perspective. Software escrow is one component of many possible solutions to minimise risk.

1.3 There is a tendency:

1.3.1 not to use software escrow agreements when ideally they should be used, taking a more strategic long term view; and

1.3.2 to treat them only as an afterthought.

1.4 Additionally, licensees, even when they use them, can fail to ensure the software escrow agreement terms are appropriate. And they can fail to manage their implementation.

2. Marketing Benefits for Suppliers

2.1 There can be real marketing benefits for software developers, particularly for new or smaller developers. Depositing source code with an escrow agent provides greater reassurance to customers that the software will be available long term. This escrow availability makes it more likely the customer will run with the local, often smaller and newer, developer rather than a Microsoft.

3. What does a Software Escrow Agreement do?

3.1 When software is licensed to a customer, the customer doesn't usually get source code (essential for software fixes, upgrades, etc.). If the developer goes under or fails to provide support, ideally the customer should have access to the source code and other material to enable it to maintain and upgrade the software for future use.

3.2 On the other hand the developer won't want to release the source code upfront, for fear that it will be misused by a licensee. For example, if the licensee makes the material available to some third party, the owner in practice loses confidentiality and its intellectual property rights.

3.3 So the two parties could appoint a trusted third party (an escrow agent) to hold a copy of the source code which is released only if, for example, the owner goes under or fails to support the software. The escrow solution will look after the interests of both parties.

3.4 The escrow agent should be someone that's neutral and independent, not a party that's aligned to one of the party's interests. They should also have relevant expertise because this job goes way beyond just storing CD-

ROMs with source code burned on them. Generally, lawyers and banks are unsuitable for this reason.

4. **Usual Agreement Format**.

4.1 Typically, the owner, the licensee, and the escrow agent sign a three way agreement (often called a tripartite agreement). This is an adjunct to the software licence agreement (which might also provide for software development although the structure works for developed software). Usually the licensee pays the escrow agent's set up, annual and ongoing costs. But the owner could pay some or all of those costs.

4.2 There will be events, such as the receivership of the owner or its failure to provide support, which trigger a process leading to supply of the source code to the licensee. When there's a triggering event (eg: a winding up petition is issued), the licensee gives notice to the escrow agent. The escrow agent or the licensee gives notice to the owner. If the owner fails to give a counter-notice within a specified time, the source code is handed across.

4.3 But if a counter-notice is given (i.e. there is a dispute as to whether a trigger event has occurred), the issue is handed over for resolution by an arbitrator or expert. Given their low fees, the software escrow agent's liability is appropriately limited. Costs in relation to disputes etc. are indemnified by the owner and licensee.

5. **Bi-Partite Escrow Agreements**

5.1 This can be a great option. More than one licensee could get the benefit of a multi-party (often called bi-partite) agreement where an owner is licensing its product to many. This can be a great use of escrow services by an emerging or smaller developer. Particularly where the software is important or critical for a customer, in deciding whether to go with that developer (or a larger multi-national such as Microsoft), customers will take into account the long term availability of the software. This involves consideration of whether the software developer will survive. From a marketing perspective, there is much to be said for a developer providing reassurance to potential customers, by setting up an escrow arrangement so source code is available if the supplier fails. The source code is held by the escrow agent. Each new licensee gets signed up. This helps local suppliers have more chance of getting selected for software projects.

5.2 How this usually works in practice is that the owner and the escrow agent sign an agreement. Then each licensee signs a document which adopts that agreement, so it gets the benefit. This can be a streamlined process, with each licensee paying only a modest sum (unless the owner pays that as well).

6. What are the issues and pitfalls?

6.1 "Off the rack" and standardised escrow services are commonly used and inexpensive. However, one size does not necessarily fit all. Particularly where risk is high, the licensee should consider tailoring the escrow services and perhaps going for a more belt-and-braces variation. This can increase cost however, so there is a cost/benefit call to be made.

6.2 Here are some of the issues.

7. What material should be lodged with the escrow agent?

7.1 Source code alone is of only limited use to a customer, faced with using and reconstructing material following default by the owner. Ideally, the material lodged with the escrow agent should include any material required to enable a reasonably skilled programmer to understand, maintain and correct the software. This could include the:

> "*product name, version or release numbers, information regarding the hardware and software environments necessary to read, compile and maintain the source code materials, details of the media type (for example, disks, magnetic tape or other media) containing the source code materials, directory listings of media content, the names and versions of development tools or libraries, software design, information, details of any encryption devices required to use the source code, any documentation required to use the source code, details of the compiler required to use the source code materials, and the names and contact details of the programmers or software engineers who wrote or created the software*". [1]

8. How can the licensee be sure all the necessary material has been deposited?

8.1 Licensees generally don't check what's been lodged. It's not unknown for blank CDs to be provided by suppliers. One escrow agent notes on its website (www.escroweurope.com) that a surprisingly high percentage of source code lodged with them fails to meet tests when evaluated. They sell 3 levels of testing. The lowest level produces a 10% failure rate. The highest 40%.

8.2 The short answer to this point is that, without detailed checking by a third party (such as a consultant employed by the escrow agent), there is not a high level of reassurance that the CD-ROM contains the right material. Of course, the licensee could take the risk and do nothing, and assume that the appropriate information has been lodged by the owner. Without paying more and getting appropriate work undertaken, escrow agents won't be checking for this.

[1] A Mitchell and A Ruggieri, *Optimising the Utility of Source Code Escrow Arrangements (Computers and Law*, January 2003 Page 26).

8.3 The reality, of course, is that, without checking, the material deposited could be useless or fall short of what is practically required. There are steps however which significantly reduce the risk but fall short of going the whole hog of fully de-compiling the source code and then running the program in a test environment. This can include sample checking of source code, anti-virus testing, checking directory listings etc. Software escrow agents often have a range of potential verification procedures available, together with pricing etc.

8.4 A key way of reducing verification cost is to require the owner to provide all necessary facilities and assistance in undertaking the verification. A suitable expert can "stand over the shoulder" of the owner's people as they run through procedures.

8.5 One unsatisfactory option is to rely on reservation of the ability to validate sometime during the life of the escrow agreement (to be called up when the licensee thinks it useful to do so). However this will almost always be too late (ie: when it's apparent that problems are about to arise).

8.6 In summary, an important judgment call is the extent to which the customer will want to have the source code and other materials checked and verified. This will be a cost/benefit question, taking into account how critical the software is to the customer's business.

9. Product Changes, Upgrades, Etc.

9.1 Under the agreement, the owner should be required to provide upgrades and other changes in accordance with the licence agreement. While an escrow agent can take modest steps to chase this, the reality is that an owner may fail to lodge upgrades. In practice, it probably falls on the licensee to chase up and make sure that the owner has lodged the material.

9.2 This emphasises the important point that escrow agreements not only must be set up well at the outset, but also that they need to be managed carefully.

10. What events trigger potential release of source code to the licensee?

10.1 There is something of a balancing act between suppliers and customers. Suppliers understandably will want to keep release of material to a minimum. Otherwise they will be excessively exposed to release of intellectual property rights and confidential material, which could be inappropriately used by customers or others. But on the other hand the customer will want quick access in as wide array of triggering events as possible, and with minimum fuss.

10.2 The typical software escrow agreement calls for the source code to go to the customer when the supplier is at or near insolvency (eg: receivership, liquidation, presentation of winding up petition etc) or if the supplier fails to perform its software maintenance and upgrading obligations.

10.3 Particularly where risk is higher for the customer, it should look at other triggering events as well, such as (a) force majeure events that go on for an extended period, and (b) when key personnel leave the developer's employment.

11. Insolvency as a Triggering Event

11.1 In practice, the owner going under is one of the major triggering events. But it is not clear cut that this will be legally enforceable. There are possible issues, such as the effect of liquidation upon commitments in a software escrow agreement, and upon property covered by that agreement. The law is unclear on this. However, even if there might be problems in theory, in practice it may well be that a liquidator or receiver will not take steps to oppose release by the escrow agent. So there is still real advantage in having the arrangements.

12. What happens when the owner and the licensee disagree about whether there has been a triggering event?

12.1 What if the owner doesn't agree with the customer's notice saying there is a triggering event? How is this resolved?

12.2 This generally should not be left to the escrow agent alone. The agent needs clarity as to when the material should be released. If the parties so ask, it should be put in the hands of a third party such as an arbitrator to resolve.

12.3 There's a problem however. Arbitration or court proceedings to resolve those issues generally take much longer than the time within which the customer really needs to get its hands on the source code.

12.4 To get a quick result, the issue must be decided by sacrificing a full and properly considered resolution by an arbitrator or a court. There are two options which usually can get a result within a month or two:

12.4.1 a shortened arbitration procedure which cuts some corners and gets a quicker result; or

12.4.2 a determination of the issue by an expert.

12.5 These two procedures are similar but the former has the strength and weaknesses of the procedure in the Arbitration Act 1996. The expert process stands outside the Arbitration Act (if set up correctly). Either way is fine as long as it encapsulates the benefits of speed and reasonable consideration of the issues, which are not just legal but often also technical as well. But the benefits of the quick-and-dirty approach come with risk of an adverse and unexpected outcome. Sometimes licensees will choose fuller arbitration in the escrow agreement even if it takes longer.

13. Who pays costs when there is a default?

13.1 Again there are different ways this can be done. But the most reasonable is that the losing party following the arbitration or expert determination should pay the costs. Of course, if the losing party is the insolvent owner, there needs to be a structure which provides for the licensee to pay the costs, including those of the escrow agent. The escrow agent's additional costs should be indemnified by the parties; their fees typically are too low to justify paying these costs themselves.

14. Escrow Agreement must integrate with main agreement

14.1 Ideally parties should check to make sure the main software development/licence agreement integrates with the escrow agreement. One example of this is to make sure that either or both agreements provide for the ability of the customer to use the source code, de-compile it etc after a triggering event (this may be excluded by the main agreement).

15. Software Escrow Variations

15.1 There are other uses for escrow, including for example deposit of code underlying website developed for the customer.

16. Third Party Software

16.1 Typically, software developed for a customer will incorporate third party software and it may be that those rights have to be dealt with as well.

17. Conclusion

17.1 Software escrow agreements are not used as much as they should be used. While not a perfect solution (they're a component of a licensee's overall business continuity planning) they do offer a way of significantly reducing risk. And they provide marketing benefits for local developers.

Wigley & Company is a specialist technology (including IT and telecommunications), procurement and marketing law firm founded 11 years ago. With broad experience in acting for both vendors and purchasers, Wigley & Company understands the issues on "both sides of the fence", and so assists its clients in achieving win-win outcomes.

While the firm acts extensively in the commercial sector, it also has a large public sector agency client base, and understands the unique needs of the public sector. While mostly we work for large organisations, we also act for SMEs.

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

The firm is actively involved in professional organisations (for example, Michael is President of the Technology Law Society and Stuart van Rij its secretary).

We welcome your feedback on this article and any enquiries you might have in respect of its contents. Please note that this article is only intended to provide a summary of the material covered and does not constitute legal advice. You should seek specialist legal advice before taking any action in relation to the matters contained in this article.