



DOING SOFTWARE ESCROWS RIGHT

Learn about the problems and solutions that exist with software escrows



By Jon Christiansen

Jon Christiansen is one of the founders of EscrowTech International, Inc., a software and technology escrow company. He is still active in the management of the company and serves as its Vice President and General Counsel. *Mr. Christiansen is a "computer lawyer" with more than 30 years of experience.*

TABLE OF CONTENTS

TABLE OF CONTENTS..... 1

WHAT THIS ARTICLE IS ABOUT..... 2

ARE SOFTWARE ESCROWS USEFUL? -
PROBLEMS AND SOLUTIONS..... 3

 THE KNOW-HOW AND BUILD PROBLEMS3

 THE FLOW-DOWN PROBLEMS..... 4

 THE UPDATE PROBLEMS 5

 CONCLUSION REGARDING SOFTWARE
 ESCROW PROBLEMS..... 6

HOW TO DO SOFTWARE ESCROWS RIGHT . 6

 IS AN ESCROW NEEDED? 6

 WHAT ABOUT SOURCE CODE LICENSES? 7

 WHO NEEDS AN ESCROW?..... 7

 WHAT SHOULD BE INCLUDED IN THE
 DEPOSIT MATERIALS?..... 7

 DEFINITION OF DEPOSIT MATERIALS 8

 ADEQUATE VS. INADEQUATE DEPOSIT
 MATERIALS 9

 ENCRYPTION OF DEPOSIT MATERIALS 9

 OWNERSHIP OF DEPOSIT MATERIALS... 10

 INTELLECTUAL PROPERTY PROTECTION
 BANKRUPTCY ACT..... 10

 MORE THAN PHYSICAL STORAGE..... 10

 HELP FROM OTHERS..... 11

 RELEASE CONDITIONS 12

 RELEASE PROCEDURES 13

 TECHNICAL VERIFICATION..... 14

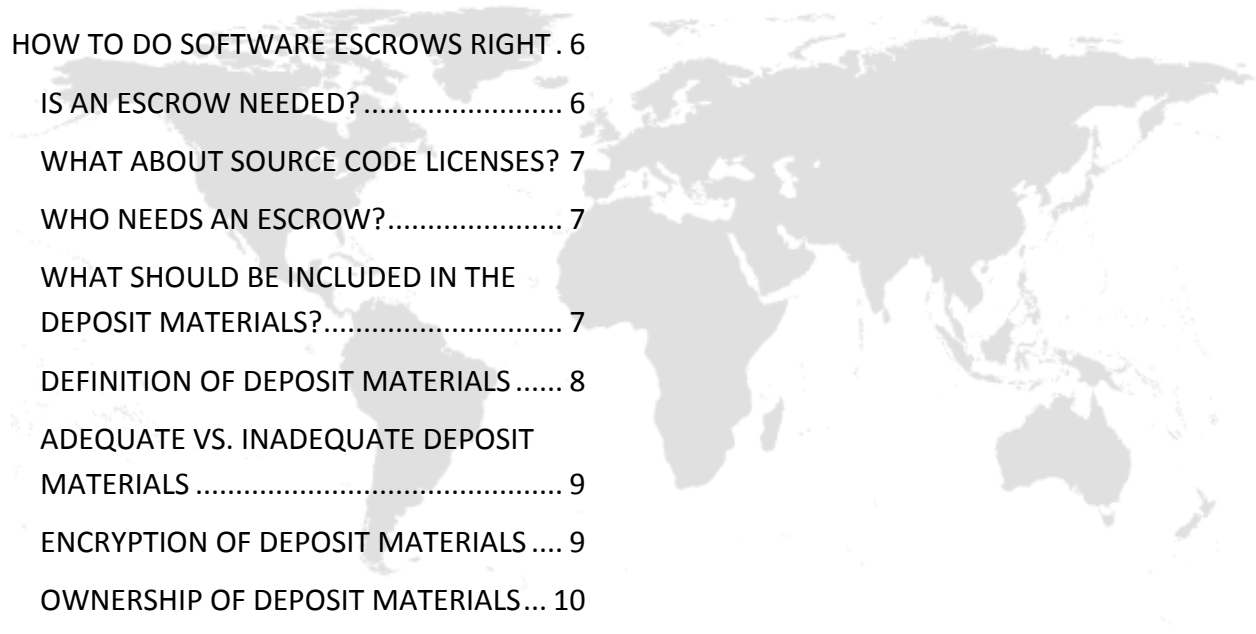
 PHYSICAL AUDIT 15

 REPRESENTATIONS FROM THE LICENSOR
 ABOUT THE DEPOSIT MATERIALS..... 15

TERMINATION 15

FEES 16

CONCLUSION..... 16



WHAT THIS ARTICLE IS ABOUT

This article takes a hard look at software escrows. Are they useful? What are some of the problems and solutions? How do you do them right?

In the interest of full disclosure, not only is the author a long time practicing "computer lawyer" and an occasional part time adjunct professor of computer law but also one of the founders of EscrowTech International, Inc., a software escrow company. This disclosure reflects both the bias and experience of the author relevant to the topic of software escrows.

Although software escrows have been used since the 1970s, they are an increasingly common practice. Today, most software lawyers deal with them occasionally, if not frequently. As an attorney, you may be asked to advise a client concerning a software escrow, or you may need to educate your client about the option of using a software escrow in a variety of situations. This article is intended to provide you with information useful for these purposes.

For a basic understanding of the purposes and mechanics of a software escrow, please take a look at "Understanding Software and Technology Escrows" at EscrowTech's website. Briefly, a software escrow protects a software licensee by ensuring that the licensee will have access to the source code (and possibly other materials) in the event

that the licensor goes out of business (e.g., via bankruptcy), discontinues support of the licensed software, breaches maintenance obligations, or some other release condition occurs. Typically, the parties use a software escrow when the license is for the object code (binary form) of the software, but the licensee does not receive the source code and therefore is unable to maintain, update, or enhance it. The licensee is dependent on the licensor for maintenance, updates, and enhancements to the software. Simplistically, a software escrow can be described as follows:

- ✓ The licensor delivers a copy of the source code to an escrow agent.
- ✓ The escrow agent holds the source code.
- ✓ The escrow agent releases the source code to the licensee only if a release condition occurs.
- ✓ The escrow agent returns the source code to the licensor if the escrow terminates without the occurrence of a release condition.

A prudent licensee will want more than just source code included in the escrow as described elsewhere in this article.

Sometimes software escrows are referred to as technology escrows. For clarification, software escrows are a subset of technology escrows. Some technology escrows do not involve software or source code. For example, technical documents, chemical formulas, prototypes, drawings, and other embodiments of intellectual

property can be held in a technology escrow. This article focuses on software escrows, but some of the concepts apply to other types of technology escrows as well.

ARE SOFTWARE ESCROWS USEFUL? - PROBLEMS AND SOLUTIONS

At least one commentator has pointed out problems associated with software escrows (and quasi-escrows that provide licensees with access to source code) and has concluded that such escrows might not be useful because of these problems. Although software escrows can pose certain problems (which is one of the reasons for this article), there are also practical solutions. The four most significant problems associated with escrows are (1) the know-how problem, (2) the build problem, (3) the flow-down problem, and (4) the update problem, each of which is discussed below, together with practical solutions.

See H. Meeker, "Thinking Outside the Lock Box: Negotiating Technology Escrows," *The Computer and Internet Lawyer*, Vol. 20, No. 9, p.6.

THE KNOW-HOW AND BUILD PROBLEMS

The licensee may not have the know-how to use the source code released from the escrow. The source code may be poorly

documented or difficult to modify. Instructions for compiling the source code and building an executable program may be needed. Accurate configuration information may be needed to configure and run the software on a specific system or with other software.

To solve these problems, the licensee and its counsel must recognize that: (1) software is a complex technology and it requires competent and skilled programmers to use source code released from an escrow; (2) the licensee might not employ computer programmers or, if it does, the programmers it employs may not possess the know-how required for the software in question; (3) source code can be poorly documented; and (4) the licensee may need build instructions, configuration information, and other programming documentation in order to make the best use of the source code.

None of these problems is difficult to overcome. If the licensee does not employ qualified computer programmers, it can hire qualified programmers as employees or independent contractors to do the job if there ever is a release of source code from the escrow. The licensee can team up with other licensees who are in the same predicament to hire and share qualified programmers or to use a legacy support service. It may even be possible to hire the same programmers who developed the software in the first place, since they may no longer have employment with the defunct licensor.

Legacy support services are sometimes established by former programmers of the licensor to maintain and support the product if it has been discontinued by the licensor or if the licensor has gone out of business. The permitted use license in an escrow agreement can and should be written to allow the licensee to take advantage of this. Other licensees without any escrow or source code rights might not be able to use this service because of intellectual property infringement issues or a hostile trustee in bankruptcy.

Furthermore, a prudent licensee can and should require that good build instructions, configuration information, and other programming documentation be included in the escrow. The licensor can be expected to make reasonable warranties about these items. These items can be inspected by the licensee or an independent contractor before they are put into escrow or by the escrow company after they are put in escrow.

THE FLOW-DOWN PROBLEMS

The software product for which the escrow is established may include third-party software that is not owned by the licensor. For example, a spell checker in a word processing program may come from a third party rather than from the licensor of the word processing program. The licensor may not have the source code for this portion (e.g., spell checker) of the product or, if it does, may not have the right to include it in the escrow or to grant a "permitted use"

license to the licensee. Thus, the escrowed materials are incomplete.

Ownership of software is an imprecise characterization. It is possible to own the copyrights and other intellectual property in software. It is also possible to own the tangible media in which software is embodied. The two are very different, and it is not clear whether ownership of software means one or both of these possibilities. In this article, ownership of software is used to mean ownership of the relevant intellectual property in the software.

Thanks to Heather Meeker for this excellent example.

A permitted use license is the license granted to the licensee for the escrowed materials if and when released to the licensee from escrow.

A licensor can solve this problem by negotiating in the agreement with the third party for the right to have third-party source code put into escrow either directly from the third party or indirectly through the licensor. If the agreement has been fully executed, the licensor can go back to the third party and request an amendment that provides the necessary rights to allow for an escrow.

This can be achieved in most cases. These third parties also want to maximize the distribution of their software because that maximizes their licensing revenues. By allowing software escrows to accommodate

the requirements of licensees, more licenses are executed and more revenue is generated. It should be noted that licensors in a flow-down situation are usually very aware of their own need for the third-party source code or for access to it if they can no longer rely on the third party, and their contracts with such third parties often already include source code rights. It is not an illogical or unreasonable jump to extend these source code rights to include an allowance for escrows that may be required by licensees.

If not, their attorneys should raise the issue with them.

If a third party still refuses to allow an escrow of third-party source code for the licensees, an escrow of the licensor's source code still has value. The bankruptcy of the licensor does not mean that the third party is gone or unavailable. The licensee will have the licensor's source code from the escrow and may be able to obtain support from the third party for its portion (e.g., spell checker) of the overall software product. From the licensee's perspective, this is better than having no source code in escrow. Moreover, it may be that the third-party portion can be replaced with a substitute. Alternatively, it may be that a licensee is better off not doing a deal in the first place with a licensor who does not have the ability to place all source code into an escrow for the protection of the licensee.

The escrow should include a permitted use license that allows the licensee to disclose source code to the third party to facilitate support of the third party's software as integrated in or with the licensor's software product.

THE UPDATE PROBLEMS

Some escrows are not properly maintained or updated. Updating an escrow may be a low priority of the licensor or may be neglected altogether. It may be difficult to verify that the licensor has deposited the necessary updates into the escrow.

To avoid this problem, the escrowed source code should be kept current with the software being used by the licensee. If the licensor does not deposit updates to keep the escrow current, then the escrow will become outdated and useless. This should never be the case for a diligent licensee, however.

The escrow agreement should require the escrow company to give written notice to the licensee each time an update is made to the escrow, and upon request, the escrow company should provide a report to the licensee of the deposits that have been made. If the licensee receives and installs a new release of the software, the licensee should expect to get a notice from the escrow company that the escrow has received updated source code. The escrow agreement should include obligations for the licensor to update the escrow and keep it current with the software licensed to the licensee. It is easy to document and monitor

updates and to know whether they have occurred. Escrow companies also can send reminders to licensors and licensees that an update has not been received within a certain period of time. The update problem is only a problem if the licensee fails to monitor the situation.

CONCLUSION REGARDING SOFTWARE ESCROW PROBLEMS

Escrows can provide a significant protection to the licensee. Software escrows are useful because they serve the business interests of both sides of the software licensing transaction. For the licensee, a competently structured and monitored escrow provides insurance by reducing and mitigating the risks and damages associated with the loss of the licensor's support and maintenance of mission-critical software.

From the perspective of the licensor, an escrow with a single escrow company with the resources and motivation to protect the escrowed source code and materials is preferable to scattering copies of such source code and materials into the hands of multiple licensees and their programmers who do not have the same resources and motivation to protect the licensor's intellectual property. Thus, an escrow is a long recognized balance between the legitimate interests of both the licensor and its licensees. Escrows help promote licensing by offering a reasonable and safe compromise to these competing interests of the licensor and its licensees.

HOW TO DO SOFTWARE ESCROWS RIGHT

This part of the article includes some suggestions on how to do a software escrow right, mostly from the perspective of the licensee that will be a beneficiary under the escrow. In this part, the licensee is frequently referred to as the "beneficiary," since this term is most commonly used to identify the person or entity entitled to receive the source code if it is released from the escrow in response to a release condition.

IS AN ESCROW NEEDED?

The first step in effecting a good escrow is to ask if an escrow is needed. Parties should not waste time, effort, and money on an escrow if it is not needed. For example, an escrow is probably not needed if the licensee is confident (and correct in its confidence) that the licensor is financially stable, will continue in business, will not discontinue maintenance and support of the software, and will not breach its maintenance or support obligations. In making this assessment, the size and fame of the licensor should not be the only consideration. Bankruptcies are not limited to small, unknown companies and individuals, and big companies can decide to drop support and maintenance of software products. An escrow is probably not needed unless the software is mission critical or of significant importance to the licensee.

The licensee should envision itself in the position of suddenly being without maintenance or support of the software because of the bankruptcy, breach, or other failure of the licensor. What then? What if the software crashes, produces erroneous results, experiences incompatibilities, or needs to be updated due to changing business needs or processes? Would it be useful to the licensee to have source code and other materials so that the licensee, or its contractors or a third-party service, could maintain and update the software for the licensee, as either a temporary or long term solution? If it is easy to quickly switch to substitute software from another source, then there may be little or no need for an escrow. If not, then an escrow may provide important protection.

These contractors or third-party service might include former employees of the licensor who are intimately familiar with the software.

WHAT ABOUT SOURCE CODE LICENSES?

A source code license is a very viable alternative to an escrow from the perspective of the licensee. If the licensor is willing to disclose and distribute its source code to its licensees, then there is no need for an escrow. An escrow is needed only for licensors that see such disclosures and distributions as an unacceptable risk. Most licensors of commercial software applications are unlikely to accept a source code license as an acceptable option, but some might. A source code license may

provide a revenue opportunity for the licensor if the licensor charges an additional fee for a source code license.

For example, many computer programs, including with source code, are available through open source licensing that eliminates the need for an escrow.

WHO NEEDS AN ESCROW?

Software escrows may be important to more than just end-user licensees. The licensee or beneficiary under an escrow might be an original equipment manufacturer (OEM), a value added reseller (VAR), a distributor, or an integrator that has a significant interest in ensuring continuity of software maintenance and support, including updates to meet future market demands. Increasingly, venture capitalists, investors, and lenders use escrows (or similar arrangements) to protect or better secure their investments in software and technology companies. Attorneys will continue to find other creative uses of software and technology escrows for a wider array of clients.

WHAT SHOULD BE INCLUDED IN THE DEPOSIT MATERIALS?

"Deposit materials" are the materials placed into the escrow. For an escrow to be effective, the beneficiary must make sure that the deposit materials are adequate. At a minimum, this means that the software's source code must be included. A prudent beneficiary will want and need more, however. Deposit materials should include build instructions, programming

documentation, configuration information, and any other documentation used by the licensor's programmers to understand the source code or to develop, compile, maintain, or update the software. Deposit materials should further include any software development tools, compilers, linkers, libraries, and other resources used by the licensor's programmers and needed for maintenance or enhancement of the software.

Generally, there is no need to include anything that would be commercially available to the beneficiary from other sources if and when deposit materials are released, however. For example, a development tool or compiler that can be obtained from a commercial source does not need to be included in the deposit materials. A list or document identifying the particular development tool or compiler, including version, should be included, though, so that the beneficiary will know what to use.

Be aware, however, that with the passage of time commercially available tools may become outdated and no longer easily available from the marketplace.

DEFINITION OF DEPOSIT MATERIALS

The actual definition or description of deposit materials used in a software escrow agreement should be drafted on a case-by-case basis after discussion with technical personnel of both the licensor and beneficiary. What is applicable or needed in

one situation may not apply in another situation. The beneficiary should make sure that the licensor is obligated to include all needed or appropriate materials. The following sample definition may need to be modified for a given situation:

The "Deposit Materials" will consist of the source code and "Development Environment" for the Software. The "Development Environment" consists of the programming documentation, build instructions, configuration information, schematics, designs, and flow charts and any proprietary software tools, libraries, linkers, utilities, compilers, and other programs used by Owner's programmers to develop, maintain or implement the Software, including instructions for compiling and linking the source code into executable forms or for building an executable version of the software. If any of the foregoing are commercial products of other companies and are readily available to Beneficiary from third party market sources, then such commercial products do not need to be included if a list identifying them is included by Owner in the Development Environment. The "Deposit Materials" will further include a list of the primary programmers involved in the development and maintenance of the Software and their home addresses and telephone numbers. This list will not be available to Beneficiary unless released as part of a release of Deposit Materials in accordance with this Software Escrow Agreement.

ADEQUATE VS. INADEQUATE DEPOSIT MATERIALS

The licensor will not want the definition of deposit materials to be nebulous, to set unreasonably high standards for the deposits, or to include items that do not exist. For example, some definitions refer to "schematics" or "designs" or a "statement of principles of operation," yet the licensor's programmers might have no idea as to what this terminology is referring.

More reasonably, the definition may require that the licensor add or improve comments to source code, create build instructions, or prepare other programming documentation that does not exist, tasks that the programmers understand but that require additional work. The beneficiary will want more than just naked source code and will want to insist on a reasonable level of build instructions and other documentation.

After proper inquiry of the technical personnel of both licensor and beneficiary, the parties' lawyers should be able to draft a definition of deposit materials that makes sense for the software in question. The driving consideration should be the reasonable need of the beneficiary in the event of a release, that is, what will the beneficiary (or its contractors) need in order to compile the source code and build and configure executable software and then to maintain and enhance the software?

This raises some interesting questions, however. Should the deposit materials be limited to whatever named materials

actually exist in their "as is" condition? Should the licensor be required to create certain documents for the escrow if they do not already exist? Should the licensor be required to improve materials that exist but do not meet certain standards or criteria? If the answer to any of these questions is yes, then who pays for the extra work that must be performed by the licensor?

There are no right or wrong answers, but it is important for lawyers to discuss these issues thoroughly with their clients. The resolution of these issues should be driven by what is reasonable under the circumstances, what level of risk the beneficiary is willing to accept, and the amount the beneficiary is willing to pay in order to reduce the risk. From the licensor's perspective, if the beneficiary wants additional or improved materials for the escrow, then the beneficiary should be willing to pay for it. From the beneficiary's perspective, the licensor owes it to its licensees/beneficiaries to have adequate materials available for the escrow. This should be part of the price of doing business as a licensor.

ENCRYPTION OF DEPOSIT MATERIALS

Generally, the deposit materials should not be in an encrypted or password-protected state, and the software escrow agreement should include a warranty to this effect from the licensor. If the licensor insists on encryption, however, the beneficiary will need access to de-encryption keys, passwords, and necessary instructions in

the event of a release. For example, the escrow company may hold the encrypted deposit materials and the beneficiary may hold the necessary de-encryption keys, passwords, and instructions.

What if the keys, passwords, or instructions given to the beneficiary are deficient or invalid, however? It does a beneficiary no good to discover this after the encrypted deposit materials are released from the escrow. Consequently, it is advisable to have the escrow company test and verify a copy of the keys, passwords, and instructions. After verification, the escrow company can destroy its copy.

OWNERSHIP OF DEPOSIT MATERIALS

The tangible deposit materials and any copies thereof made by the escrow company in accordance with the escrow agreement should be owned by the escrow company, but such ownership should not include any copyrights or other intellectual property in or to the deposit materials. As the owner of the tangible deposit materials (e.g., CD-ROMs in which source code is stored), the escrow company is in a better position to release or withhold deposit materials in accordance with its contractual obligations under the escrow agreement. For example, in the event of the licensor's bankruptcy and a trustee's rejection of the license agreement and escrow agreement as executory contracts, it should be clear that the tangible deposit materials are owned by the escrow company, not the licensor or its estate.

INTELLECTUAL PROPERTY PROTECTION BANKRUPTCY ACT

Congress passed the Intellectual Property Protection Bankruptcy Act to ensure that a licensee of intellectual property receives the benefit of its bargain, even after a licensee's bankruptcy. To take full advantage of the Act, the escrow agreement should include statements similar to the following:

This Software Escrow Agreement is intended by the Parties to be "supplementary" to the License Agreement within the meaning of Section 365(n) of the US Bankruptcy Code (11 U.S.C. 365(n)). If this Agreement and/or the License Agreement is (are) rejected by Licensor as a debtor in possession or a trustee or by any other person or entity under the US Bankruptcy Code, then Beneficiary may elect to retain its rights as provided in Section 365(n).

The Deposit Materials are an "embodiment" of "intellectual property" as those terms are used in Section 365(n) of the US Bankruptcy Code (11 U.S.C. 365(n)).

MORE THAN PHYSICAL STORAGE

It is a common mistake to view a software escrow as merely an arrangement for the physical storage and disclosure/transfer of source code and other materials to a beneficiary in the event a release condition occurs. Although this is a critical purpose of the escrow, there is a second purpose that is just as important to the beneficiary. A proper escrow will provide the legal

structure necessary for the beneficiary to use the source code and other deposit materials after they are released to the licensee.

For example, the escrow agreement should include a permitted use license that permits the beneficiary to use the source code to modify the software for maintenance purposes, to compile the modified source code, and to use and copy the maintained/modified/compiled software as permitted in the original license agreement. In some cases, the permitted use license may need to be expanded to include the right to enhance and update the source code and software to keep the software current and useful, the right to create derivative works beyond maintenance purposes, the right to distribute the source code to others, and the right for the licensee to use independent contractors or third party support services for these purposes.

Normally there would be no right to distribute or publish source code, but there can be exceptions, e.g., a beneficiary that is a VAR or OEM may need to provide source code to its customers, especially if the VAR or OEM discontinues business or maintenance of the software.

The specific needs of the beneficiary will dictate the appropriate scope of the permitted use license. The license should be broad enough to meet the reasonable needs of the beneficiary, but not unreasonably broad from the perspective of

the licensor. The permitted use license should be granted at the time that the escrow is entered into so that the beneficiary can elect to retain the license under the Intellectual Property Bankruptcy Protection Act in the event of a subsequent bankruptcy by the licensor. It is important that the license be granted prior to the filing of the bankruptcy petition so that the beneficiary can elect to retain the permitted use license in the event that the license agreement is rejected by a trustee or debtor-in-possession. As a practical matter, the timing of this grant is not a real threat to the licensor since the source code is physically held by the escrow company and the escrow company is under a contractual obligation not to release the source code until after the release condition. Even then, the escrow company must comply with the release procedure of the escrow agreement.

For a similar reason, the escrow agreement should include a simple license granted by the licensor to the escrow company to copy and release the source code and other escrowed materials and to otherwise take any action needed for the escrow. This way, the escrow company can also elect to retain its license in the event the trustee or debtor-in-possession rejects the escrow agreement.

HELP FROM OTHERS

In the event of a release of deposit materials, the beneficiary is usually not the only one left high and dry; there are usually other licensees. The escrow agreement

(e.g., in the permitted use license) should allow the beneficiary to team up with these other licensees and to jointly engage an independent contractor with the requisite expertise and experience to use the source code in accordance with the permitted use license, and to do so for the benefit of all of the licensees. This permits a sharing of costs, resources, and expertise. The independent contractor may even be a maintenance service that hires former programmers of the licensor who are intimately familiar with the source code and its build/development environment. The beneficiary should ensure that the permitted use licensee is broad enough to allow for all the above.

RELEASE CONDITIONS

In drafting an escrow agreement, the release conditions (sometimes called trigger events) are just as important as the deposit materials. The occurrence of a release condition entitles the beneficiary to receive the deposit materials. There may be one or more release conditions. Here are just a few simple examples of release conditions used in escrow agreements:

- ✓ Licensor discontinues business because of insolvency or bankruptcy, and no successor assumes licensor's software maintenance obligations under the license agreement.
- ✓ Licensor (or its successor) defaults in its obligation to provide maintenance services as required by the license agreement and fails to

cure such default within two weeks after receiving written notice of the default from beneficiary. The notice must describe the default and the action that beneficiary believes is necessary to cure the default. If more than two weeks is reasonably required to complete the cure, licensor (or its successor) shall have such additional time (not to exceed two months) as is reasonably needed, provided that licensor (or its successor) is diligent in completing the cure.

- ✓ Licensor ceases to maintain the software for beneficiary while under a maintenance obligation to beneficiary, and no successor to licensor continues to maintain the software for beneficiary.
- ✓ Licensor discontinues its business relating to the software and no successor to such business assumes and carries out licensor's contractual obligations to maintain the software for beneficiary.
- ✓ Licensor becomes insolvent or admits either insolvency or a general inability to pay its debts as they become due.
- ✓ Licensor files a petition for protection under the US Bankruptcy Code or an involuntary petition in bankruptcy is filed against licensor and is not dismissed within 60 days thereafter.
- ✓ Licensor refuses or fails to renew its maintenance and support

obligations under the license agreement after beneficiary has requested such renewal in a writing delivered to licensor.

- ✓ Licensor becomes the subject of a Chapter 7 bankruptcy proceeding under the US Bankruptcy Code, and such proceeding is not dismissed within 90 days after its initiation.
- ✓ Licensor is acquired by or merges with a competitor of beneficiary.

Many other (including much more detailed) release conditions are possible. Release conditions are a matter of agreement between the licensor and the beneficiary, in consultation with their professional and legal advisors. Generally, escrow companies do not care how the release conditions are defined. Escrow companies are much more concerned about the release procedures described below.

RELEASE PROCEDURES

The escrow agreement should spell out the release procedure for a release of deposit materials to the beneficiary. Typically, the release procedure is either (1) a procedure with a waiting period and a right for the licensor to object or (2) a procedure for a mandatory or immediate release.

The first release procedure is the one most commonly used but is the least favorable to the beneficiary. The first procedure can generally be described as follows: If the beneficiary believes that a release condition has occurred, the beneficiary will deliver a written request or demand to the escrow

company that the deposit materials be released. The escrow company will not release the deposit materials at that time. Instead, the escrow company will notify the licensor.

The licensor will have a waiting period (e.g., two weeks, but this frequently varies) within which to object to the release if the licensor believes that the release condition did not occur. If the escrow company receives the objection within the waiting period, then the escrow company will hold the deposit materials and the dispute will be resolved by litigation or arbitration or other agreed upon dispute resolution method. The escrow company will release or continue to hold deposit materials as decided by the court or arbitrator. The licensor usually does not object, presumably because it knows that the beneficiary is correct in its allegation that the release condition has occurred or because the licensor is out of business and no one is left to object or care. Objections can and do occur on occasion, however.

The second release procedure provides for a mandatory and immediate release of the deposit materials. There is no waiting period and there is no opportunity for the licensor to object to the release before it is made. The licensor is typically given an opportunity to challenge the validity of the release through litigation or arbitration, however, and may seek an order from the court or arbitrator that the deposit materials be returned to escrow. If successful, the licensor is generally entitled

to recover its expenses, including attorneys' fees.

A beneficiary has a strong incentive to want this mandatory release procedure. If a release condition occurs, the beneficiary may be damaged by the delay of a waiting period even if there is no objection from the licensor. The beneficiary would be even further damaged by the delay of litigation or arbitration if the licensor is entitled to object. Beneficiaries should carefully evaluate the timing associated with a release procedure. In some cases, time may be of the essence in securing a release.

A licensor may be concerned about a mandatory release procedure, however, because a beneficiary might incorrectly claim that the release condition has occurred and might receive deposit materials without justification. The licensor's concern is legitimate but should not be overstated. A properly drafted escrow agreement will include in or with the permitted use license strict confidentiality provisions and other restrictions binding on the beneficiary with respect to any use, copying, or disclosure of the deposit materials. Therefore, even under a mandatory release, the beneficiary could not lawfully distribute or publish the source code to the world or otherwise exceed the limits of the permitted use license. If a mandatory release procedure is not acceptable to the licensor, a shorter waiting period and provisions for an expedited arbitration to resolve a dispute, if any, may be a good compromise.

TECHNICAL VERIFICATION

For especially critical or important software, the beneficiary may desire to have the escrow company perform technical verification services to verify the content of the deposit materials. This answers the beneficiary's question: "How do I know that proper and complete source code has been deposited into the escrow?" To reassure the beneficiary, the deposit materials may be verified by having the escrow company compile the source code and create or build an executable application. The application (in binary or object code form) can then be run against a test plan or provided to the beneficiary for testing to ensure that it matches the software used by the beneficiary. This process can be very involved and expensive. The verification may take place at the facilities of the escrow company or of the licensor.

Lesser levels of verification are possible and are less expensive. For example, having a technical verification specialist of the escrow company or an independent contractor of the beneficiary observe the preparation of deposit materials before they are deposited into escrow can be useful. This enables a "look over the shoulder" of the licensor and may include useful suggestions on build instructions, configuration information, tools, and documentation that should be included in the deposit materials. When source code for mission-critical or highly important software is escrowed, the benefits of technical verification generally justify the

cost, even though such costs can be much higher than the escrow fees. For attorneys representing beneficiaries, the question of technical verification should always be discussed.

PHYSICAL AUDIT

Even if the escrow company does not conduct a technical verification, the beneficiary will want to make sure that the escrow company conducts a physical audit of the deposit materials each time they are received. The escrow company should not blindly accept sealed containers or packages from the licensor but should compare the received materials against an inventory list. The escrow company also should check CD-ROMs (or other media) to make sure they are not damaged and that they are readable. Sometimes a CD-ROM is missing or is on a bad medium and cannot be read. The escrow company should also scan the CD-ROMs for viruses. The escrow company should check to see if the materials are encrypted. If so, are the necessary de-encryption keys or passwords included? None of this provides the same level of certainty as a technical verification, but it is a simple and important level of effort that should be required of the escrow company.

REPRESENTATIONS FROM THE LICENSOR ABOUT THE DEPOSIT MATERIALS

Although not a substitute for technical verification, the escrow agreement should include a requirement that the licensor include with each deposit of materials a

written description of those materials and that the escrow agent provide a copy of such written description to the beneficiary. The escrow agreement should also include a provision to the effect that such descriptions are representations from the licensor that the deposit materials conform to those descriptions. In this manner, the licensor is at least required to represent the content of each deposit.

TERMINATION

An often-overlooked issue is termination of the escrow. The escrow should terminate if the corresponding license agreement (or the license granted through the agreement) expires or terminates. What about licenses that are perpetual or for a term of 99 years, however? From the perspective of the licensor, the escrow should terminate if and when the maintenance and support obligations of the licensor expire or terminate. Generally, there is no justification for a beneficiary to receive source code from an escrow if the beneficiary is no longer entitled to maintenance and support from the licensor.

As a related issue, both the licensor and beneficiary should make sure that the escrow agreement does not commit them to use a particular escrow company for any extended period of time. They should be able to fire the escrow company at any time and replace it with a substitute escrow company.

FEES

The question inevitably arises as to who should pay the escrow fees, the licensor or the beneficiary. There is no right answer or standard practice. In a majority of cases, it is the licensor who is responsible for payment of the escrow fees to the escrow company. This is viewed by many as an expected cost of doing business. In many of these cases, however, the beneficiary reimburses the licensor or even pays a greater fee to the licensor to cover not only the escrow fees but also the licensor's administrative expenses. There are many situations in which the beneficiary pays the escrow company directly. For licensors, it may be best to offer the option of an escrow (for payment of an escrow fee) when license fees and other fees are first disclosed to a prospective licensee. This way, the licensor can make it clear to the prospective licensee that if the licensee wants an escrow, it will be required to pay for it.

CONCLUSION

With a little knowledge, care and effort, it is relatively easy to structure an effective software escrow that provides meaningful protection to a licensee without undue risk or burden on the licensor. A beneficiary who follows the above suggestions is not likely to be disappointed in the effectiveness of an escrow.

Contact EscrowTech at (801) 852-8202 for more information about Software Escrows.