



## SUGGESTED ESCROW MATERIALS

The following list contains recommended materials to be deposited into an escrow account beyond just source code. Please note that this list is not meant to be all-inclusive and we strongly advise you to have a thorough discussion with your organization’s technology professionals and legal staff in order to identify the materials that are specific to your needs. Your technology developer, the organization who will be depositing into the account, may also be unable or unwilling to deposit certain materials due to legal restrictions. Your escrow agreement should fully detail and outline the limits in place by either party. If you have any questions or concerns please feel free to contact your Iron Mountain representative.

ITEM/INFORMATION REQUESTED	COMMENTS
<b>GENERAL INFORMATION</b>	
Roadmap of Deposit Materials	A high level overview describing where key items are located on the deposit media will save your organization precious time if the deposit is released!
Information regarding: (1) deposit media type, (2) Type/density (tapes only), (3) drive manufacturer (tapes only), (4) utilities, or third party applications used to create tapes; as well as (5) non – default parameters used in creating the tapes that would affect the extraction process	<ol style="list-style-type: none"> <li>1. CD/DVD, Secure FTP, Tape</li> <li>2. E.g. LTO-5</li> <li>3. Quantum, IBM, etc.</li> <li>4. tar, cpio, Dell Netvault , Symantec NetBackup, etc.</li> <li>5. Modified block size, etc.</li> </ol>
Encryption information, required passwords, and/or crypto-keys or software programs required to access the deposit materials.	<p>e.g. WinZip, WinRAR, PGP, openSSL, etc. – version number should be included.</p> <p>Any necessary passwords or keys should be provided to Iron Mountain at the same time as the source code deposit - most reliably and easily in a separate deposit that can be shipped independently if security is a concern.</p>
Source Code	<p>Varies depending on programming language(s) used; examples of common file types: ‘.java’, ‘.c’, ‘.cpp’, ‘.cs’, ‘.vb’ etc.</p> <p>Any files or items that are required to be present in the build environment in order to successfully compile that are not installed to the build environment by either the installation of the OS or any third party tools should be considered source code whether it falls into the commonly accepted types of “source code” or not.</p>

<p>Tools/information required in order to produce a clear-text version of the source code or any other unique characteristics of the deposit that should be known regarding the extraction of the data.</p>	<p>If the software will be deposited in the form of a software configuration management system repository (e.g. Team Foundation Server, Git, Subversion, etc.) information required may include such items as SCM tool version, host operating system(s), specific import/export parameters, etc.</p> <p>Preferably software should be checked out/extracted from any Configuration Management tool prior to deposit, however if this is not possible (e.g. build scripts are heavily integrated with the SCM tool), the deposit should also contain information identifying label or versioning information necessary to check out the correct version of the software for the deposit account beneficiary. The labels or other specific global identifiers associated with specific release versions should be provided.</p>
<p>Contact information for key programmers including, but not limited to, address, phone numbers, etc.</p>	
<p>Component dependencies</p>	<p>Often in order to compile or link, one software component requires another to have been built/compiled first. A list identifying all known dependencies between components is extremely helpful.</p>
<p>Build Environment setup and configuration</p>	<p>Detailed instructions for configuration of build environment; including, but not limited to, specification of hardware environments and operating system(s) including version and patch/service pack levels, configuration settings for all compilers and third party components. For the purposes of an escrow account the build environment setup documentation should be tailored to the escrow deposit and any references in generally available documentation to machines or shares available only on the depositor's local network should be resolved relative to the escrow deposit.</p>
<p>Build control files</p>	<p>The files and/or scripts that control the build process. For example, Makefiles on UNIX systems or '.sln', '.csproj', '.vbproj', etc. in Visual Studio. Highly variable depending upon OS and specific tools used.</p> <p>This should also include any batch or shell scripts that are used to control the build of the software.</p>
<p>Any third party applications, libraries, API's or other tools, whether included in the escrow deposit or not, necessary to compile and build executable code, objects, dynamic libraries, etc. [please note any required service packs or patches/updates]</p>	<p>May include compilers, linkers, third party libraries, assemblers, pre-processors, post-processors, linkers, whether developed by Depositor or by third parties.</p> <p>Note: if these materials can not be deposited due to licensing restrictions, the Depositor should provide commercial/retail product names of all required tools and applications, version numbers – including any patch or service pack numbers, vendor name and contact information (including website) of all required items.</p>

Build Instructions	<p>Detailed instructions/sequence of actions describing how to compile the escrow deposit and build executables once the correct build environment has been configured. These can sometimes be as simple as loading a single Visual Studio Solution file into the VS IDE and building or if the build is more manual then much more specific information should be provided. If builds are performed from a command line using tools such as 'make' or 'ant', any specific build targets required should be identified. If specific batch or shell scripts must be run to perform a build these should be identified with their location and any specific order of execution and parameter requirements.</p> <p>Keep in mind that the person doing the build an escrow release situation will, with near certainty, have never run a build in this specific environment before so care should be taken to ensure that no 'institutional knowledge' is assumed in the build instructions as the person performing the build will not have this knowledge nor any reliable means to acquire it.</p>
Design Documentation	<p>Documentation providing a reasonably skillful programmer with information regarding source code architecture, overall design of the course code and interactions among modules.</p> <p>While design documentation is usually never 100% up-to-date with current software development any documentation is better than none for a developer who would be tasked with maintaining the software should a release condition occur.</p>
API's/program interface documentation	<p>Specifications for any API's that are exposed by the application(s). Any documentation used by developers during design, code, test or maintenance phase of the software life-cycle.</p>
Test Diagnostics	<p>Regression tests including automated scripts and test data if available.</p>
<b>DATA/DATABASE INFORMATION</b>	
Data	<p>Sample/examples of any data or databases required to compile or run the application(s). May be data used for regression testing.</p>
Database Schema/Data model documents	<p>Details of database schema and design information.</p> <p>If appropriate the deposit should contain the necessary SQL files to create and/or backup files required to configure a database suitable for supporting compilation of the software and/or supporting the run-time function of the software</p>
<b>RUNTIME/PRODUCTION INFORMATION</b>	

<p>Third party libraries, dll's, applications, scripts (xml, html, etc.) and data required to establish the "live" environment</p>	<p>A detailed listing of all exe's, third party libraries, data and test diagnostics. While these items may not be available for deposit due to licensing issues, at a minimum, the Depositor should provide names of all required applications, version number – with patch or service pack number, vendor name and contact information (including website) of all required items.</p>
<p>Hosting Configuration Instructions</p>	<p>Detailed steps necessary to configure environment hosting executable code – may not be known/available to the Beneficiary due to licensing issues.</p> <p>If the application is distributed via a SaaS or similar model where the beneficiary does not host the application on their own machines this is especially critical and should be as detailed as possible.</p>
<p>Network Design Information</p>	
<p>Runtime/Production Environment configuration instructions</p>	<p>Information regarding operating systems, diagnostics; including instructions regarding installation of the application and steps required to load the application onto a production/runtime environment. This should include identifying all third party components required for production system as well as any configuration settings. Again this is especially critical for those situations where the beneficiary does not host the application on their own machines.</p>
<p>User Manuals/Training guides</p>	

**CONFIDENTIAL AND PROPRIETARY INFORMATION OF IRON MOUNTAIN**