PKWARE, Inc., Third-Party License Summary for PKZIP® and SecureZIP® for AIX, HP-UX, Solaris, Linux and Windows Server v14.0

The PKWARE® PKZIP® and SecureZIP® for AIX, HP-UX, Solaris, Linux and Windows Server v14.0 product makes use of software modules or code developed by one or more third parties. This document identifies the licensing information governing this third-party software.

Contents

4.4BSD-Lite

Below is given the disclaimer for 4.4BSD-Lite.

The 4.4BSD and 4.4BSD-Lite software is distributed under the following terms:

All of the documentation and software included in the 4.4BSD and 4.4BSD-Lite Releases is copyrighted by The Regents of the University of California.

Copyright 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994 The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

   This product includes software developed by the University of California, Berkeley and its contributors.
4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The Institute of Electrical and Electronics Engineers and the American National Standards Committee X3, on Information Processing Systems have given us permission to reprint portions of their documentation.

In the following statement, the phrase “this text” refers to portions of the system documentation.

Portions of this text are reprinted and reproduced in electronic form in the second BSD Networking Software Release, from IEEE Std 1003.1-1988, IEEE Standard Portable Operating System Interface for Computer Environments (POSIX), copyright C 1988 by the Institute of Electrical and Electronics Engineers, Inc. In the event of any discrepancy between these versions and the original IEEE Standard, the original IEEE Standard is the referee document.

In the following statement, the phrase "This material" refers to portions of the system documentation.

This material is reproduced with permission from American National Standards Committee X3, on Information Processing Systems. Computer and Business Equipment Manufacturers Association (CBEMA), 311 First St., NW, Suite 500, Washington, DC 20001-2178. The developmental work of Programming Language C was completed by the X3J11 Technical Committee.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of the Regents of the University of California.

NOTE: The copyright of UC Berkeley's Berkeley Software Distribution ("BSD") source has been updated. The copyright addendum may be found at ftp://ftp.cs.berkeley.edu/pub/4bsd/README.Impt.License.Change and is included below.

July 22, 1999

To All Licensees, Distributors of Any Version of BSD:

As you know, certain of the Berkeley Software Distribution ("BSD") source code files require that further distributions of products containing all or portions of the software, acknowledge within their advertising materials that such products contain software developed by UC Berkeley and its contributors.

Specifically, the provision reads:

3. *All advertising materials mentioning features or use of this softwaremust display the following acknowledgement: This product includes software developed by the University ofCalifornia, Berkeley and its contributors.*

Effective immediately, licensees and distributors are no longer required to include the acknowledgement within advertising materials. Accordingly, the foregoing paragraph of those BSD Unix files containing it is hereby deleted in its entirety.

William Hoskins
Director, Office of Technology Licensing
University of California, Berkeley

7-Zip

SecureZIP for Windows Command Line uses parts of the 7-Zip program to support this file type. You can find the source code at http://www.7-zip.org. Below are given the license and disclaimer for 7Zip.

```
  7-Zip
  ~~~~~
  License for use and distribution
  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

  7-Zip Copyright (C) 1999-2011 Igor Pavlov.

  Licenses for files are:

    1) 7z.dll: GNU LGPL + unRAR restriction
    2) All other files:  GNU LGPL

  The GNU LGPL + unRAR restriction means that you must follow both
  GNU LGPL rules and unRAR restriction rules.


  Note:
    You can use 7-Zip on any computer, including a computer in a
commercial organization. You don't need to register or pay for 7-Zip.


  GNU LGPL information
  --------------------

    This library is free software; you can redistribute it and/or
    modify it under the terms of the GNU Lesser General Public
    License as published by the Free Software Foundation; either
    version 2.1 of the License, or (at your option) any later version.

    This library is distributed in the hope that it will be useful,
    but WITHOUT ANY WARRANTY; without even the implied warranty of
    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
    Lesser General Public License for more details.

    You can receive a copy of the GNU Lesser General Public License
from http://www.gnu.org/


  unRAR restriction
  -----------------

    The decompression engine for RAR archives was developed using
source code of unRAR program.
    All copyrights to original unRAR code are owned by Alexander
Roshal.
    The license for original unRAR code has the following restriction:
```

The unRAR sources cannot be used to re-create the RAR compression algorithm, which is proprietary. Distribution of modified unRAR sources in separate form or as a part of other software is permitted, provided that it is clearly stated in the documentation and source comments that the code may not be used to develop a RAR (WinRAR) compatible archiver.

--
Igor Pavlov

BZIP2

Below are given the license and disclaimer for BZIP2.

CRC-32

Below are given the license and disclaimer for CRC-32.

ଔ

crc32.c -- compute the CRC-32 of a data stream

Copyright (C) 1995-2003 Mark Adler

For conditions of distribution and use, see copyright notice in zlib.h Thanks to Rodney Brown <rbrown64@csc.com.au> for his contribution of faster CRC methods: exclusive-oring 32 bits of data at a time, and pre-computing tables for updating the shift register in one step with three exclusive-ors instead of four steps with four exclusive- ors. This results about a factor of two increase in speed on a Power PC G4 (PPC7455) using gcc -O3.

zlib.h -- interface of the 'zlib' general purpose compression library version 1.1.3, July 9th, 1998

ଔ

Copyright (C) 1995-1998 Jean-loup Gailly and Mark Adler

This software is provided ‘as-is’, without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly Mark Adler
jloup@gzip.org madler@alumni.caltech.edu

The data format used by the zlib library is described by RFCs (Request for Comments) 1950 to 1952 in the files ftp://ds.internic.net/rfc/rfc1950.txt (zlib format), rfc1951.txt (deflate format) and rfc1952.txt (gzip format).

ଔ

LZMA

SecureZIP implements the LZMA SDK under the SPECIAL EXCEPTION licensing terms described below.

ଔ

LZMA SDK 4.57

LZMA SDK Copyright (C) 1999-2007 Igor Pavlov

LZMA SDK provides the documentation, samples, header files, libraries, and tools you need to develop applications that use LZMA compression.

LZMA is default and general compression method of 7z format in 7-Zip compression program (www.7-zip.org). LZMA provides high compression ratio and very fast decompression.

LZMA is an improved version of famous LZ77 compression algorithm. It was improved in way of maximum increasing of compression ratio, keeping high decompression speed and low memory requirements for decompressing.

LICENSE

LZMA SDK is available under any of the following licenses:

1) GNU Lesser General Public License (GNU LGPL)
2) Common Public License (CPL)
3) Simplified license for unmodified code (read SPECIAL EXCEPTION)
4) Proprietary license

It means that you can select one of these four options and follow rules of that license.

1,2) GNU LGPL and CPL licenses are pretty similar and both these licenses are classified as

- "Free software licenses" at http://www.gnu.org/
- "OSI-approved" at http://www.opensource.org/

3) SPECIAL EXCEPTION

Igor Pavlov, as the author of this code, expressly permits you to statically or dynamically link your code (or bind by name) to the files from LZMA SDK without subjecting your linked code to the terms of the CPL or GNU LGPL. Any modifications or additions to files from LZMA SDK, however, are subject to the GNU LGPL or CPL terms.

SPECIAL EXCEPTION allows you to use LZMA SDK in applications with closed code, while you keep LZMA SDK code unmodified.

SPECIAL EXCEPTION #2: Igor Pavlov, as the author of this code, expressly permits you to use this code under the same terms and conditions contained in the License Agreement you have for any previous version of LZMA SDK developed by Igor Pavlov.

SPECIAL EXCEPTION #2 allows owners of proprietary licenses to use latest version of LZMA SDK as update for previous versions.

SPECIAL EXCEPTION #3: Igor Pavlov, as the author of this code, expressly permits you to use code of the following files: BranchTypes.h, LzmaTypes.h, LzmaTest.c, LzmaStateTest.c, LzmaAlone.cpp, LzmaAlone.cs, LzmaAlone.java as public domain code.

4) Proprietary license

1) Right to modify code without subjecting modified code to the terms of the CPL or GNU LGPL

2) Technical support for code

To request such proprietary license or any additional consultations, send email message from that page:

http://www.7-zip.org/support.html

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA You should have received a copy of the Common Public License along with this library.

LZMA SDK Contents

LZMA SDK includes:

- C++ source code of LZMA compressing and decompressing

- ANSI-C compatible source code for LZMA decompressing

- C# source code for LZMA compressing and decompressing

- Java source code for LZMA compressing and decompressing

- Compiled file->file LZMA compressing/decompressing program for Windows system ANSI-C LZMA decompression code was ported from original C++ sources to C. Also it was simplified and optimized for code size. But it is fully compatible with LZMA from 7-Zip.

UNIX/Linux version

To compile C++ version of file->file LZMA, go to directory

C/7zip/Compress/LZMA_Alone

and type "make" or "make clean all" to recompile all.

In some UNIX/Linux versions you must compile LZMA with static libraries.

To compile with static libraries, change string in makefile

LIB = -lm

to string

LIB = -lm -static

Files

```
C        - C source code
CPP      - CPP source code
CS       - C# source code
Java     - Java source code
lzma.txt - LZMA SDK description (this file)
7zFormat.txt - 7z Format description
7zC.txt  - 7z ANSI-C Decoder description (this file)
methods.txt  - Compression method IDs for .7z
LGPL.txt - GNU Lesser General Public License
CPL.html - Common Public License
lzma.exe - Compiled file->file LZMA encoder/decoder for Windows
history.txt - history of the LZMA SDK
```

Source code structure

C - C files

```
    Compress - files related to compression/decompression
      Lz     - files related to LZ (Lempel-Ziv) compression algorithm
      Lzma   - ANSI-C compatible LZMA decompressor

        LzmaDecode.h  - interface for LZMA decoding on ANSI-C
        LzmaDecode.c      - LZMA decoding on ANSI-C (new fastest
version)
        LzmaDecodeSize.c  - LZMA decoding on ANSI-C (old size-optimized
version)
        LzmaTest.c        - test application that decodes LZMA encoded
file
        LzmaTypes.h       - basic types for LZMA Decoder
        LzmaStateDecode.h - interface for LZMA decoding (State version)
        LzmaStateDecode.c - LZMA decoding on ANSI-C (State version)
        LzmaStateTest.c   - test application (State version)

      Branch       - Filters for x86, IA-64, ARM, ARM-Thumb, PowerPC
and SPARC code

    Archive - files related to archiving
      7z_C     - 7z ANSI-C Decoder
```

CPP -- CPP files

```
  Common  - common files for C++ projects
  Windows - common files for Windows related code
  7zip    - files related to 7-Zip Project

    Common   - common files for 7-Zip
```

```
    Compress - files related to compression/decompression

      LZ     - files related to LZ (Lempel-Ziv) compression algorithm

      Copy         - Copy coder
      RangeCoder   - Range Coder (special code of
compression/decompression)
      LZMA         - LZMA compression/decompression on C++
      LZMA_Alone   - file->file LZMA compression/decompression

      Branch       - Filters for x86, IA-64, ARM, ARM-Thumb, PowerPC
and SPARC code

    Archive - files related to archiving

      Common   - common files for archive handling
      7z       - 7z C++ Encoder/Decoder

    Bundles    - Modules that are bundles of other modules

      Alone7z           - 7zr.exe: Standalone version of 7z.exe that
supports only 7z/LZMA/BCJ/BCJ2
      Format7zR         - 7zr.dll: Reduced version of 7za.dll:
extracting/compressing to 7z/LZMA/BCJ/BCJ2
      Format7zExtractR  - 7zxr.dll: Reduced version of 7zxa.dll:
extracting from 7z/LZMA/BCJ/BCJ2.

    UI        - User Interface files

      Client7z - Test application for 7za.dll,  7zr.dll, 7zxr.dll
      Common   - Common UI files
      Console  - Code for console archiver
```

CS - C# files

```
  7zip
    Common   - some common files for 7-Zip
    Compress - files related to compression/decompression
      LZ     - files related to LZ (Lempel-Ziv) compression algorithm
      LZMA         - LZMA compression/decompression
      LzmaAlone    - file->file LZMA compression/decompression
      RangeCoder   - Range Coder (special code of
compression/decompression)
```

Java - Java files

```
  SevenZip
    Compression    - files related to compression/decompression
      LZ           - files related to LZ (Lempel-Ziv) compression
algorithm
      LZMA         - LZMA compression/decompression
      RangeCoder   - Range Coder (special code of
compression/decompression)

C/C++ source code of LZMA SDK is part of 7-Zip project.
```

You can find ANSI-C LZMA decompressing code at folder

C/7zip/Compress/Lzma

7-Zip doesn't use that ANSI-C LZMA code and that code was developed specially for this SDK. And files from C/7zip/Compress/Lzma do not need files from other directories of SDK for compiling.

7-Zip source code can be downloaded from 7-Zip's SourceForge page:

http://sourceforge.net/projects/sevenzip/

LZMA features

- Variable dictionary size (up to 1 GB)
- Estimated compressing speed: about 1 MB/s on 1 GHz CPU
- Estimated decompressing speed:
  - 8-12 MB/s on 1 GHz Intel Pentium 3 or AMD Athlon
  - 500-1000 KB/s on 100 MHz ARM, MIPS, PowerPC or other simple RISC
- Small memory requirements for decompressing (8-32 KB + DictionarySize)
- Small code size for decompressing: 2-8 KB (depending from speed optimizations)

LZMA decoder uses only integer operations and can be implemented in any modern 32-bit CPU (or on 16-bit CPU with some conditions).

Some critical operations that affect to speed of LZMA decompression:

1) 32*16 bit integer multiply

2) Misspredicted branches (penalty mostly depends from pipeline length)

3) 32-bit shift and arithmetic operations

Speed of LZMA decompressing mostly depends from CPU speed.

Memory speed has no big meaning. But if your CPU has small data cache, overall weight of memory speed will slightly increase.

How To Use

```
Using LZMA encoder/decoder executable
-------------------------------------

Usage:  LZMA <e|d> inputFile outputFile [<switches>...]

  e: encode file

  d: decode file

  b: Benchmark. There are two tests: compressing and decompressing
     with LZMA method. Benchmark shows rating in MIPS (million
     instructions per second). Rating value is calculated from
     measured speed and it is normalized with AMD Athlon 64 X2 CPU
     results. Also Benchmark checks possible hardware errors (RAM
     errors in most cases). Benchmark uses these settings:
     (-a1, -d21, -fb32, -mfbt4). You can change only -d. Also you
```

```
     can change number of iterations. Example for 30 iterations:
       LZMA b 30
     Default number of iterations is 10.

<Switches>


  -a{N}:  set compression mode 0 = fast, 1 = normal
          default: 1 (normal)

  d{N}:   Sets Dictionary size - [0, 30], default: 23 (8MB)
          The maximum value for dictionary size is 1 GB = 2^30 bytes.
          Dictionary size is calculated as DictionarySize = 2^N bytes.
          For decompressing file compressed by LZMA method with
dictionary
          size D = 2^N you need about D bytes of memory (RAM).

  -fb{N}: set number of fast bytes - [5, 273], default: 128
          Usually big number gives a little bit better compression
ratio
          and slower compression process.

  -lc{N}: set number of literal context bits - [0, 8], default: 3
          Sometimes lc=4 gives gain for big files.

  -lp{N}: set number of literal pos bits - [0, 4], default: 0
          lp switch is intended for periodical data when period is
          equal 2^N. For example, for 32-bit (4 bytes)
          periodical data you can use lp=2. Often it's better to set
lc0,
          if you change lp switch.

  -pb{N}: set number of pos bits - [0, 4], default: 2
          pb switch is intended for periodical data
          when period is equal 2^N.

  -mf{MF_ID}: set Match Finder. Default: bt4.
              Algorithms from hc* group doesn't provide good
compression
              ratio, but they often works pretty fast in combination
with
              fast mode (-a0).

              Memory requirements depend from dictionary size
              (parameter "d" in table below).

               MF_ID     Memory                   Description

                 bt2    d *  9.5 + 4MB  Binary Tree with 2 bytes
hashing.
                 bt3    d * 11.5 + 4MB  Binary Tree with 3 bytes
hashing.
                 bt4    d * 11.5 + 4MB  Binary Tree with 4 bytes
hashing.
                 hc4    d *  7.5 + 4MB  Hash Chain with 4 bytes hashing.

  -eos:   write End Of Stream marker. By default LZMA doesn't write
```

```
          eos marker, since LZMA decoder knows uncompressed size
          stored in .lzma file header.

  -si:    Read data from stdin (it will write End Of Stream marker).
  -so:    Write data to stdout


Examples:

1) LZMA e file.bin file.lzma -d16 -lc0

compresses file.bin to file.lzma with 64 KB dictionary (2^16=64K)
and 0 literal context bits. -lc0 allows to reduce memory requirements
for decompression.


2) LZMA e file.bin file.lzma -lc0 -lp2

compresses file.bin to file.lzma with settings suitable
for 32-bit periodical data (for example, ARM or MIPS code).

3) LZMA d file.lzma file.bin

decompresses file.lzma to file.bin.
```

Compression ratio hints

```
Recommendations
---------------

To increase compression ratio for LZMA compressing it's desirable to
have aligned data (if it's possible) and also it's desirable to locate
data in such order, where code is grouped in one place and data is
grouped in other place (it's better than such mixing: code, data, code,
data, ...).


Using Filters
-------------
You can increase compression ratio for some data types, using special
filters before compressing. For example, it's possible to increase
compression ratio on 5-10% for code for those CPU ISAs:

x86, IA-64, ARM, ARM-Thumb, PowerPC, SPARC.

You can find C/C++ source code of such filters in folder
"7zip/Compress/Branch"

You can check compression ratio gain of these filters with such
7-Zip commands (example for ARM code):

No filter:
  7z a a1.7z a.bin -m0=lzma

With filter for little-endian ARM code:
  7z a a2.7z a.bin -m0=bc_arm -m1=lzma
```

With filter for big-endian ARM code (using additional Swap4 filter):
  7z a a3.7z a.bin -m0=swap4 -m1=bc_arm -m2=lzma

It works in such manner:
Compressing   = Filter_encoding + LZMA_encoding
Decompressing = LZMA_decoding + Filter_decoding

Compressing and decompressing speed of such filters is very high, so it will not increase decompressing time too much. Moreover, it reduces decompression time for LZMA_decoding, since compression ratio with filtering is higher.

These filters convert CALL (calling procedure) instructions from relative offsets to absolute addresses, so such data becomes more compressible. Source code of these CALL filters is pretty simple (about 20 lines of C++), so you can convert it from C++ version yourself.

For some ISAs (for example, for MIPS) it's impossible to get gain from such filter.

LZMA compressed file format

Offset	Size	Description
0	1	Special LZMA properties for compressed data
1	4	Dictionary size (little endian)
5	8	Uncompressed size (little endian). -1 means unknown size
13		Compressed data

ANSI-C LZMA Decoder

To compile ANSI-C LZMA Decoder you can use one of the following files sets:
1) LzmaDecode.h + LzmaDecode.c + LzmaTest.c (fastest version)
2) LzmaDecode.h + LzmaDecodeSize.c + LzmaTest.c (old size-optimized version)
3) LzmaStateDecode.h + LzmaStateDecode.c + LzmaStateTest.c (zlib-like interface)

Memory requirements for LZMA decoding

LZMA decoder doesn't allocate memory itself, so you must allocate memory and send it to LZMA.

Stack usage of LZMA decoding function for local variables is not larger than 200 bytes.

How To decompress data

LZMA Decoder (ANSI-C version) now supports 5 interfaces:
1) Single-call Decompressing
2) Single-call Decompressing with input stream callback

```
3) Multi-call Decompressing with output buffer
4) Multi-call Decompressing with input callback and output buffer
5) Multi-call State Decompressing (zlib-like interface)

Variant-5 is similar to Variant-4, but Variant-5 doesn't use callback
functions.

Decompressing steps
-------------------

1) read LZMA properties (5 bytes):
   unsigned char properties[LZMA_PROPERTIES_SIZE];

2) read uncompressed size (8 bytes, little-endian)

3) Decode properties:

  CLzmaDecoderState state;  /* it's 24-140 bytes structure, if int is
32-bit */

  if (LzmaDecodeProperties(&state.Properties, properties,
LZMA_PROPERTIES_SIZE) != LZMA_RESULT_OK)
    return PrintError(rs, "Incorrect stream properties");

4) Allocate memory block for internal Structures:

  state.Probs = (CProb *)malloc(LzmaGetNumProbs(&state.Properties) *
sizeof(CProb));
  if (state.Probs == 0)
    return PrintError(rs, kCantAllocateMessage);

  LZMA decoder uses array of CProb variables as internal structure. By
default, CProb is unsigned_short. But you can define _LZMA_PROB32 to
make it unsigned_int. It can increase speed on some 32-bit CPUs, but
memory usage will be doubled in that case.


5) Main Decompressing

You must use one of the following interfaces:

5.1 Single-call Decompressing
-----------------------------
When to use: RAM->RAM decompressing
Compile files: LzmaDecode.h, LzmaDecode.c
Compile defines: no defines
Memory Requirements:
  - Input buffer: compressed size
  - Output buffer: uncompressed size
  - LZMA Internal Structures (~16 KB for default settings)

Interface:
  int res = LzmaDecode(&state,
      inStream, compressedSize, &inProcessed,
      outStream, outSize, &outProcessed);
```

```
5.2 Single-call Decompressing with input stream callback
-------------------------------------------------------
When to use: File->RAM or Flash->RAM decompressing.
Compile files: LzmaDecode.h, LzmaDecode.c
Compile defines: _LZMA_IN_CB
Memory Requirements:
  - Buffer for input stream: any size (for example, 16 KB)
  - Output buffer: uncompressed size
  - LZMA Internal Structures (~16 KB for default settings)

Interface:
  typedef struct _CBuffer
  {
    ILzmaInCallback InCallback;
    FILE *File;
    unsigned char Buffer[kInBufferSize];
  } CBuffer;

  int LzmaReadCompressed(void *object, const unsigned char **buffer,
SizeT *size)
  {
    CBuffer *bo = (CBuffer *)object;
    *buffer = bo->Buffer;
    *size = MyReadFile(bo->File, bo->Buffer, kInBufferSize);
    return LZMA_RESULT_OK;
  }

  CBuffer g_InBuffer;

  g_InBuffer.File = inFile;
  g_InBuffer.InCallback.Read = LzmaReadCompressed;
  int res = LzmaDecode(&state,
      &g_InBuffer.InCallback,
      outStream, outSize, &outProcessed);


5.3 Multi-call decompressing with output buffer
-----------------------------------------------
When to use: RAM->File decompressing
Compile files: LzmaDecode.h, LzmaDecode.c
Compile defines: _LZMA_OUT_READ
Memory Requirements:
 - Input buffer: compressed size
 - Buffer for output stream: any size (for example, 16 KB)
 - LZMA Internal Structures (~16 KB for default settings)
 - LZMA dictionary (dictionary size is encoded in stream properties)

Interface:

  state.Dictionary = (unsigned char
*)malloc(state.Properties.DictionarySize);

  LzmaDecoderInit(&state);
  do
  {
    LzmaDecode(&state,
      inBuffer, inAvail, &inProcessed,
```

```
      g_OutBuffer, outAvail, &outProcessed);
    inAvail -= inProcessed;
    inBuffer += inProcessed;
  }
  while you need more bytes

  see LzmaTest.c for more details.


5.4 Multi-call decompressing with input callback and output buffer
------------------------------------------------------------------
When to use: File->File decompressing
Compile files: LzmaDecode.h, LzmaDecode.c
Compile defines: _LZMA_IN_CB, _LZMA_OUT_READ
Memory Requirements:
 - Buffer for input stream: any size (for example, 16 KB)
 - Buffer for output stream: any size (for example, 16 KB)
 - LZMA Internal Structures (~16 KB for default settings)
 - LZMA dictionary (dictionary size is encoded in stream properties)

Interface:

  state.Dictionary = (unsigned char
*)malloc(state.Properties.DictionarySize);

  LzmaDecoderInit(&state);
  do
  {
    LzmaDecode(&state,
      &bo.InCallback,
      g_OutBuffer, outAvail, &outProcessed);
  }
  while you need more bytes

  see LzmaTest.c for more details:


5.5 Multi-call State Decompressing (zlib-like interface)
--------------------------------------------------------
When to use: file->file decompressing
Compile files: LzmaStateDecode.h, LzmaStateDecode.c
Compile defines:
Memory Requirements:
 - Buffer for input stream: any size (for example, 16 KB)
 - Buffer for output stream: any size (for example, 16 KB)
 - LZMA Internal Structures (~16 KB for default settings)
 - LZMA dictionary (dictionary size is encoded in stream properties)

Interface:

  state.Dictionary = (unsigned char
*)malloc(state.Properties.DictionarySize);


  LzmaDecoderInit(&state);
  do
  {
```

```
    res = LzmaDecode(&state,
      inBuffer, inAvail, &inProcessed,
      g_OutBuffer, outAvail, &outProcessed,
      finishDecoding);
    inAvail -= inProcessed;
    inBuffer += inProcessed;
  }
  while you need more bytes

  see LzmaStateTest.c for more details:


6) Free all allocated blocks


Note
----
LzmaDecodeSize.c is size-optimized version of LzmaDecode.c. But
compiled code of LzmaDecodeSize.c can be larger than compiled code of
LzmaDecode.c. So it's better to use LzmaDecode.c in most cases.


EXIT codes
-----------

LZMA decoder can return one of the following codes:

#define LZMA_RESULT_OK 0
#define LZMA_RESULT_DATA_ERROR 1

If you use callback function for input data and you return some error
code, LZMA Decoder also returns that code.



LZMA Defines
------------

_LZMA_IN_CB    - Use callback for input data

_LZMA_OUT_READ - Use read function for output data

_LZMA_LOC_OPT  - Enable local speed optimizations inside code.
                 _LZMA_LOC_OPT is only for LzmaDecodeSize.c (size-
optimized version).
                 _LZMA_LOC_OPT doesn't affect LzmaDecode.c (speed-
optimized version)
                 and LzmaStateDecode.c

_LZMA_PROB32   - It can increase speed on some 32-bit CPUs,
                 but memory usage will be doubled in that case

_LZMA_UINT32_IS_ULONG  - Define it if int is 16-bit on your compiler
                         and long is 32-bit.

_LZMA_SYSTEM_SIZE_T  - Define it if you want to use system's size_t.
                       You can use it to enable 64-bit sizes supporting
```

C++ LZMA Encoder/Decoder
~~~~~~~~~~~~~~~~~~~~~~~
C++ LZMA code use COM-like interfaces. So if you want to use it, you can study basics of COM/OLE.

By default, LZMA Encoder contains all Match Finders. But for compressing it's enough to have just one of them. So for reducing size of compressing code you can define:

```
  #define COMPRESS_MF_BT
  #define COMPRESS_MF_BT4
```

and it will use only bt4 match finder.

---

http://www.7-zip.org
http://www.7-zip.org/support.html
~~~~~~~~~~~~~~~~~~~~~~~

Net-SNMP

Below are given the license and disclaimer for Net-SNMP.

Various copyrights apply to this package, listed in various separate parts below. Please make sure that you read all the parts. Up until 2001, the project was based at UC Davis, and the first part covers all code written during this time. From 2001 onwards, the project has been based at SourceForge, and Networks Associates Technology, Inc hold the copyright on behalf of the wider Net-SNMP community, covering all derivative work done since then. An additional copyright section has been added as Part 3 below also under a BSD license for the work contributed by Cambridge Broadband Ltd. to the project since 2001. An additional copyright section has been added as Part 4 below also under a BSD license for the work contributed by Sun Microsystems, Inc. to the project since 2003.

Code has been contributed to this project by many people over the years it has been in development, and a full list of contributors can be found in the README file under the THANKS section.

Part 1: CMU/UCD copyright notice: (BSD like)

Copyright 1989, 1991, 1992 by Carnegie Mellon University

Derivative Work - 1996, 1998-2000

Copyright 1996, 1998-2000 The Regents of the University of California

All Rights Reserved

Permission to use, copy, modify and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of CMU and The Regents of the University of California not be used in advertising or publicity pertaining to distribution of the software without specific written permission.

CMU AND THE REGENTS OF THE UNIVERSITY OF CALIFORNIA DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL CMU OR THE REGENTS OF THE UNIVERSITY OF CALIFORNIA BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM THE LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

---- Part 2: Networks Associates Technology, Inc copyright notice (BSD) -----

Copyright (c) 2001-2003, Networks Associates Technology, Inc All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Networks Associates Technology, Inc nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Part 3: Cambridge Broadband Ltd. copyright notice (BSD)

Portions of this code are copyright (c) 2001-2003, Cambridge Broadband Ltd. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* The name of Cambridge Broadband Ltd. may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDER “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Part 4: Sun Microsystems, Inc. copyright notice (BSD)

Copyright © 2003 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Use is subject to license terms below.

This distribution may include materials developed by third parties.

Sun, Sun Microsystems, the Sun logo and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Sun Microsystems, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Part 5: Sparta, Inc copyright notice (BSD)

Copyright (c) 2003-2005, Sparta, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Sparta, Inc nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Part 6: Cisco/BUPTNIC copyright notice (BSD)

Copyright (c) 2004, Cisco, Inc and Information Network Center of Beijing University of Posts and Telecommunications. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Cisco, Inc, Beijing University of Posts and Telecommunications, nor the names of their contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;

LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Part 7: Fabasoft R&D Software GmbH & Co KG copyright notice (BSD)

Copyright (c) Fabasoft R&D Software GmbH & Co KG, 2003 oss@fabasoft.com

Author: Bernhard Penz <bernhard.penz@fabasoft.com>

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* The name of Fabasoft R&D Software GmbH & Co KG or any of its subsidiaries, brand or product names may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDER “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

OpenLDAP Public License

The software includes portions that are copyrighted by the OpenLDAP Foundation, 1998-2003, and are used under the OpenLDAP Public License. This is used only within UNIX/Linux/FreeBSD implementions of PKWARE software.

ଔ

The OpenLDAP Public License

Version 2.7, 7 September 2001

Redistribution and use of this software and associated documentation (“Software”), with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain copyright statements and notices,
2. Redistributions in binary form must reproduce applicable copyright statements and notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution, and
3. Redistributions must contain a verbatim copy of this document.

The OpenLDAP Foundation may revise this license from time to time. Each revision is distinguished by a version number. You may use this Software under terms of this license revision or under the terms of any subsequent revision of the license.

THIS SOFTWARE IS PROVIDED BY THE OPENLDAP FOUNDATION AND ITS CONTRIBUTORS “AS IS” AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OPENLDAP FOUNDATION, ITS CONTRIBUTORS, OR THE AUTHOR(S) OR OWNER(S) OF THE SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The names of the authors and copyright holders must not be used in advertising or otherwise to promote the sale, use or other dealing in this Software without specific, written prior permission. Title to copyright in this Software shall at all times remain with copyright holders.

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

ଔ

OpenSSL Public License

Below are given the license and disclaimer for OpenSSL.

ꕥ

License Issues

The OpenSSL toolkit stays under a dual license; that is, both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually, both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL, please contact openssl-core@openssl.org.

OpenSSL License

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment:

   "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.
5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment:

   "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)

   THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED

AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Original SSLeay License

Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com) All rights reserved.

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are aheared to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement: “This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)” The word ‘cryptographic’ can be left out if the rouines from the library being used are not cryptographic related :-).
4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

PPMd

PPMd is used under the terms described below.

ଔ

1. Short DESCRIPTION.

PPMd program is file-to-file compressor, it is written for embedding in user programs mainly and it is not intended for immediate use. I was interested in speed and performance improvements of abstract PPM model [1-6] only, without tuning it to particular data types, therefore compressor works good enough for texts, but it is not so good for nonhomogeneous files (executables) and for noisy analog data (sounds, pictures etc.). Program is very memory consuming, You can choose balance between execution speed and memory economy, on one hand, and compression performance, on another hand, with the help of model order selection option (-o).

Methods of restoration of model correctness at memory insufficiency:

'-r0 - restart model from scratch'. This method is not optimal for any type of data sources, but it works fast and efficient in average, so it is the recommended method.

'-r1 - cut off model'. This method is optimal for quasistationary sources when the period of stationarity is much larger than period between cutoffs. As a rule, it gives better results, but it is slower than other methods and it is unstable against fragmentation of memory heap at high model orders and low memory.

'-r2 - freeze model'. This method is optimal for stationary sources (show me such source when You will find it ;-)). It is fast and efficient for such sources.

2. Distribution CONTENTS.

read_me.txt - this file;
PPMd.h, PPMdType.h - header files;
Coder.hpp, SubAlloc.hpp, Model.cpp, PPMd.cpp - code sources;
makefile.gmk - makefile for GnuC v.2.95.2 (tested for DJGPP v.2.03 only);
makefile.imk - makefile for IntelC v.4.0;
makefile.mak - makefile for BorlandC v.5.01;
PPMd.exe - compressor itself (2.097bpb on Calgary corpus);
PPMonstr.exe - fat and sleepy compressor (for taking PPMd down only, 1.963bpb on Calgary corpus);

3. LEGAL stuff.

You can not misattribute authorship on algorithm or code sources, You can not patent algorithm or its parts, all other things are allowed and welcomed.

Dmitry Subbotin and me have authorship rights on code sources. Dmitry Subbotin owns authorship rights on his variation of rangecoder algorithm and I own authorship rights on my variation of PPM

algorithm. This variation is named PPMII (PPM with Information Inheritance).

PPMonstr program is distributed for experiments and noncommercial use only.

4. DIFFERENCES between variants.

Jun 13, 1999 var.A
Initial release;
Jun 30, 1999 var.B
Arithmetic coder was changed to newer version;
Simplified LOE was tested (Model1.cpp file);
Some small improvements were done;
Aug 22, 1999 var.C
Rudimentary SEE was added;
Some small improvements were done;
Oct 6, 1999 var.D
Inherited probabilities (IPs) were added;
Memory requirements were reduced a bit;
Small improvements were continued;
Dec 3, 1999 var.E
Program name was changed from PPM, escape method D (PPMD) to PPM by Dmitry (PPMd). Pronounce correctly! ;-)
Bug in ARI_FLUSH_ENCODER was crushed;
Model1.cpp file was removed from package due to LOE gives no gain;
Apr 7, 2000 var.F(inal?)
Michael Schindler`s rangecoder implementation was replaced with 'carryless rangecoder' by Dmitry Subbotin. Now, PPMd is pure public domain program;
CC rate - 2.123/2.089bpb;
Nov 26, 2000 var.G(rand final)
Memory requirements were reduced;
CC rate - 2.121/2.056bpb;
Apr 21, 2001 var.H(ard run to final)
Memory requirements were reduced a bit;
CC rate - 2.104/2.041bpb;
Apr 28, 2002 var.I(t is final too)
References to papers were corrected;
CC rate - 2.097/1.963bpb;
Apr 30, 2002 var.I rev.1
One defect in PPMonstr was fixed, this revision of PPMonstr is not compatible with previous one;

5. REFERENCES.

[1] Excellent introductory review T.Bell, I.H.Witten, J.G.Cleary 'MODELING FOR TEXT COMPRESSION'. Russian translation is placed at http://cotty.mebius.net/compress/ru/modeling.txt;

[2] Very descriptive M.R.Nelson`s COMP-2 program (PPMd is based on it). COMP-2 is in wuarchive.wustl.edu:/mirrors/msdos/ddjmag/ddj9102.zip (inner zip file nelson.zip);

[3] P.G.Howard PhD thesis 'The Design and Analysis of Efficient Lossless Data Compression Systems', is available in ftp.cs.brown.edu/pub/techreports/93/cs93-28.ps.Z;

[4] S.Bunton PhD thesis 'On-Line Stochastic Processes in Data Compression', is available in ftp.cs.washington.edu/tr/1997/03/UW-CSE-97-03-02.PS.Z;

PPMII algorithm:

[5] D.Shkarin 'Improving the efficiency of PPM algorithm', in Russian, http://sochi.net.ru/~maxime/doc/PracticalPPM.ps.gz;

[6] D.Shkarin 'PPM: one step to practicality', in English, http://www.dogma.net/DataCompression/Miscellaneous/PPMII_DCC02.pdf

Dancy compression!

Dmitry Shkarin
E-mail: dmitry.shkarin@mtu- net.ru

SQLite

Below are given the license and disclaimer for software written for SQLite. This is used only within UNIX/Linux/FreeBSD implementions of PKWARE software.

—— ☙❧

All of the deliverable code in SQLite has been dedicated to the public domain by the authors. All code authors, and representatives of the companies they work for, have signed affidavits dedicating their contributions to the public domain and originals of those signed affidavits are stored in a firesafe at the main offices of Hwaci. Anyone is free to copy, modify, publish, use, compile, sell, or distribute the original SQLite code, either in source code form or as a compiled binary, for any purpose, commercial or non-commercial, and by any means.

The previous paragraph applies to the deliverable code in SQLite - those parts of the SQLite library that you actually bundle and ship with a larger application. Portions of the documentation and some code used as part of the build process might fall under other licenses. The details here are unclear. We do not worry about the licensing of the documentation and build code so much because none of these things are part of the core deliverable SQLite library.

All of the deliverable code in SQLite has been written from scratch. No code has been taken from other projects or from the open internet. Every line of code can be traced back to its original author, and all of those authors have public domain dedications on file. So the SQLite code base is clean and is uncontaminated with licensed code from other projects.

——☙❧

UnRAR

Below are given the license and disclaimer for software written by Alexander Roshal for implementing decompression for RAR.

ଔ

The source code of UnRAR utility is freeware. This means:

1. All copyrights to RAR and the utility UnRAR are exclusively owned by the author - Alexander Roshal.

2. The UnRAR sources may be used in any software to handle RAR archives without limitations free of charge, but cannot be used to re-create the RAR compression algorithm, which is proprietary. Distribution of modified UnRAR sources in separate form or as a part of other software is permitted, provided that it is clearly stated in the documentation and source comments that the code may not be used to develop a RAR (WinRAR) compatible archiver.

3. The UnRAR utility may be freely distributed. It is allowed to distribute UnRAR inside of other software packages.

4. THE RAR ARCHIVER AND THE UnRAR UTILITY ARE DISTRIBUTED "AS IS".
NO WARRANTY OF ANY KIND IS EXPRESSED OR IMPLIED. YOU USE AT YOUR OWN RISK. THE AUTHOR WILL NOT BE LIABLE FOR DATA LOSS, DAMAGES, LOSS OF PROFITS OR ANY OTHER KIND OF LOSS WHILE USING
OR MISUSING THIS SOFTWARE.

5. Installing and using the UnRAR utility signifies acceptance of these terms and conditions of the license.

6. If you don't agree with terms of the license you must remove UnRAR files from your storage devices and cease to use the utility.

ଔ

WavPack

Below is given the disclaimer for WavPack