

FathZIP component

By Fath Software

License agreement

This Limited Use Software License Agreement (the "Agreement") is a legal agreement between you, the end-user ("Licensee"), and author. By using this software or storing this program ("FathZIP component") on a computer hard drive or other media, you are agreeing to be bound by the terms of this Agreement.

You may install trial version of this program to test and evaluate for 30 days; after that time you must either register this program or delete it from your computer hard drive.

The trial version of software may be distributed freely on online services, bulletin boards, or other electronic media as long as the files are distributed in their entirety.

You may not alter this software in any way, including changing or removing any messages or windows.

You may not decompile, reverse engineer, disassemble or otherwise reduce this software to a human perceivable form. You may not modify, rent or resell for profit this software. You may not publicize or distribute any registration code algorithms, information, or registration codes used by this software without permission of author.

Author grants a license to use the enclosed software to the original purchaser.

Customer written applications containing embedded FathZIP component may be freely distributed, without royalty payments to author, provided that such distributed product is bound into these applications in such a way so as to prohibit separate use in design mode, and that such product is distributed only in conjunction with the customer's own software product.

This control may be used as a constituent control only if the compound control thus created is distributed with and as an integral part of an application. This license may be transferred to a third party only if all existing copies of the software and its documentation are also transferred.

This product is licensed for use by only one developer at a time. Author expressly prohibits installing this product on more than one computer if there is any chance that both copies will be used simultaneously. This restriction also extends to installation on a network server, if more than one workstation will be accessing the product. All developers working on a project which includes an FathZIP component product, even though not working directly with the product, are required to purchase a license for that product.

This software is provided "as is". Author makes no warranty, expressed or implied, with regard to the software. All implied warranties, including the warranties of merchantability and fitness for a particular use, are hereby excluded.

AUTHOR'S LIABILITY IS LIMITED TO THE PURCHASE PRICE. Under no circumstances shall author of this product be liable for any incidental or consequential damages, nor for any damages in excess of the original purchase price.

Plain English version:

We require that you purchase one copy of a control per developer on a project. If this is met, you may distribute the control with your application royalty free. You may never distribute the LIC file. You may not change the product in any way that removes or changes the requirement of a license file.

We encourage the use of our controls as constituent controls when the compound controls you create are an integral part of your application. But we don't allow distribution of our controls as constituents of other controls when the compound control is not part of an application. The reason we need to have this restriction is that without it someone might decide to use our control as a constituent, add some trivial (or even non-trivial) enhancements and then sell the compound control. Obviously there would be little difference between that and just plain reselling our control.

If you have purchased the source code, you may not re-distribute the source code either (nor may you copy it into your own project). Author retains the copyright to the source code.

Your license is transferable. The original purchaser of the product must make the transfer request. Contact us for further information.

The sample versions of our products are intended for evaluation purposes only. You may not use the sample version to develop completed applications.

Technical support

Internet Mail

You can send E-mail to technical support via the Internet. Messages should be addressed to support@fathsoft.com .

World Wide Web

The Fath Software web site is located at <http://www.fathsoft.com> . You can access our web site for up-to-date information about the product, support forum, knowledgebase and news.

FathZIP component reference

BasePath	Specify the starting path for compressing files
CompressionLevel	Controls the amount of compression to be applied to files being zipped
ExclusiveMode	If set to TRUE, AddFile method locks the source file while zipping
FileAttribute	Returns file system attributes of file in archive
FileComment	Reads file comment string
FileCount	Returns number of files in archive
FileDate	Returns date of file in archive
Filename	Returns name of file in archive
FileSize	Returns size of file in archive
FileSizeZ	Returns compressed size of file in archive
FilesProcessed	Returns number of files affected by last operation
GlobalComment	Reads global comment string
IsCrypted	Returns crypted flag for file in archive
LastError	Returns last error number
Password	Password to use for encrypting / decrypting files
PreservePaths	Control whether paths get saved into ZIP
ProcessSubfolders	Control whether subfolders are processed
Version	Returns version string
VolumeSize	Set span volume size
AddFile	Add file to archive
AddFileFromMemory	Add file from memory to archive
BrowseForFolder	Show Browse-For-Folder dialog
Close	Close archive
CreateZip	Create archive
ExtractFile	Extract file from archive
ExtractFile2	Extract file from archive with new filename
ExtractToMemory	Extract file from archive to memory
ExtractToMemoryArray	Extract file from archive and return VARIANT(BYTE ARRAY)
GzCopy	Uncompress GZ file
IsFullVersion	Returns TRUE if full version of FathZIP is in use
MakeSFX	Converts ZIP file into self-extracting archive
OpenZip	Open archive for extracting and/or adding files
RemoveFile	Remove file from archive
SetFileComment	Sets comment text for a file in the archive
SetGlobalComment	Sets global archive comment text
TestFile	Test integrity of the file in archive

ChangeDisk	A new disk (or span file) is required while zipping/unzipping spanned archives
DataIn	Provide you with the data of a file that is being unzipped to memory.
GetData	Event is triggered so that you can provide the data of a file that is being zipped from memory.
OverallStatus	Reports compression/decompression overall progress
Status	Reports compression/decompression progress
UnzipPreprocessFile	The UnzipPreprocessFile event provides the opportunity to view and change a file's information before it is unzipped.
ZipPreprocessFile	The ZipPreprocessFile event provides the opportunity to view and change a file's information before it is zipped into archive.

Constants

Properties

BasePath

Property BasePath As String

Specify the starting path for compressing files

The BasePath property allows you to specify the starting path for all filepaths in the AddFile and ExtractFile methods that have been specified using relative paths.

If the BasePath property is left empty, relative paths specified in the AddFile and ExtractFile methods will be relative to the application's current directory, which is not recommended.

This property is particularly useful for [controlling how paths are stored in the zip file](#).

CompressionLevel

Property CompressionLevel As Long

Controls the amount of compression to be applied to files being zipped

Possible values

0 No compression. Files are stored in the zip file as raw data.

1 Minimum compression. This setting takes the least amount of time to compress files. When compared to the 6 setting, this setting usually gives noticeably faster compression times in exchange for about 1 to 15% larger compressed files.

6 Normal compression. This is the best balance between the time it takes to compress a file and the compression ratio achieved (default).

9 Maximum compression. This setting achieves the best compression ratios that the Zip deflate compression algorithm is capable of producing. When compared to the 6 setting, this setting significantly increases compression time in exchange for about 1-3% smaller compressed files. We recommend that you use this setting only when you really need to achieve the smallest possible files and when compression time is unimportant.

ExclusiveMode

Property ExclusiveMode As Long

If set to TRUE, AddFile method locks the source file while zipping

FileAttribute

Property FileAttribute (rec As Long) As Long

Returns file system attributes of file in archive

Rec argument is an index of file in archive in range 0 to FileCount - 1 .

Call this method to get file system attributes of file stored in ZIP.
Before call to this method, open archive with OpenZip.

FileComment

Property FileComment (rec As Long) As String

Reads file comment string

FileCount

Property FileCount As Long

Returns number of files in archive

Call this method to number of files in archive.
Before call to this method, open archive with OpenZip.
When you finish, call Close to close archive file.

FileDate

Property FileDate (rec As Long) As Date

Returns date of file in archive

Call this method to dateof file in archive.
Before call to this method, open archive with OpenZip.
When you finish, call Close to close archive file.

Filename

Property Filename (rec As Long) As String

Returns name of file in archive

Call this method to get filenames from archive.
Before call to this method, open archive with OpenZip.
When you finish, call Close to close archive file.

FileSize

Property FileSize (rec As Long) As Long

Returns size of file in archive

FileSizeZ

Property FileSizeZ (rec As Long) As Long

Returns compressed size of file in archive

FilesProcessed

Property FilesProcessed As Long

Returns number of files affected by last operation

The FilesProcessed property returns number of files affected by last AddFile or ExtractFile operation.

GlobalComment

Property GlobalComment As String

Reads global comment string

Maximum string length is 2048 characters.
Archive must be previously opened with OpenZip method.

IsCrypted

Property IsCrypted (rec As Long) As Long

Returns crypted flag for file in archive

LastError

Property LastError As Long

Returns last error number

The LastError property returns number of the error that last occurred while executing some of the FathZip methods.

Password

Property Password As String

Password to use for encrypting / decrypting files

The Password property contains the case-sensitive password to be used for encrypting or decrypting files. The password specified in the Password property applies to all the files being zipped, unzipped or tested.

If you don't need ZIP encryption, set this property to an empty string (default).

PreservePaths

Property PreservePaths As Long

Control whether paths get saved into ZIP

The PreservePaths property allows you to determine whether path information is stored in (or retrieved) from the zip file.

When using the AddFile method, setting PreservePaths to False will cause all path information associated with a file to be discarded, and will cause only the file's filename to be stored in the zip file. Setting PreservePaths to True will cause path information to be stored in the zip file along with each file's filename. See the controlling how paths are stored topic for more information on storing paths when using the AddFile method.

When using the ExtractFile method, setting PreservePaths to False will cause FathZIP to ignore any path information stored in the zip file.

Also when using the ExtractFile method: If PreservePaths is set to True, pathnames that have been stored in the zip file along with each file's filename will be used to determine where the files will be unzipped to. The stored pathnames and files will be created inside the destination unzipping folder specified with the path argument. If PreservePaths is set to False, all files will be directly unzipped (without any subfolders being created) to the destination unzipping folder.

Default value for this property is TRUE.

See [controlling how paths are stored in the zip file](#).

ProcessSubfolders

Property ProcessSubfolders As Long

Control whether subfolders are processed

The ProcessSubfolders property is used to determine whether subfolders are processed or not.

When the ProcessSubfolders property is set to True, you can zip or unzip entire folder structures. For example: Specifying "C:\WINDOWS*" as the AddFile argument and setting the ProcessSubfolders property to True will cause the method to zip up the entire contents of the "C:\WINDOWS" folder, including the SYSTEM folder and all other files and folders that can be found in there.

When the ProcessSubfolders property is set to False, subfolders will not be processed.

Default value for this property is TRUE.

Use "*" to match all files in a given folder. Old-style DOS wildcards such as "*.*" do not work in 32-bit operating systems (Windows 95/98/NT) the way they do in DOS. In Windows 95/98/NT, "*.*" will only match filenames having a "." .

See [controlling how paths are stored in the zip file](#).

Version

Property Version As String

Returns version string

VolumeSize

Property VolumeSize As Long

Set span volume size

See CreateZip method.

Methods

AddFile

Sub AddFile (Filename As String, Comment As String)

Add file to archive

Arguments

Filename is a full path to file(s) you want to compress.
Comment is an ZIP file local comment for this file.

Call this method to add file to archive. Wildcards (* and ?) can be used to compress whole directory or directory tree.

Before call to this method, create archive with CreateZip or open it with OpenZip.
When you finish, call Close to close archive file.
While compressing Status event is invoked.
Use | to delimit multiple file names.

LastError values

0 No error
-1 File not found

```
AddFile("c:\folder\*.doc | c:\folder2\*.txt", "")
```

AddFileFromMemory

Sub AddFileFromMemory (Filename As String, FileSize As Long, FileDate As Date, FileAttrib As Long, FileComment As String)

Add file from memory to archive

Compress memory data to a file in ZIP.
Handle GetData event to provide data for a file that is being zipped.

Arguments

Filename is a name of file in archive.
FileSize is a size of the file.
FileDate is a date to store with file in ZIP.
FileAttrib is a file-system attributes to store with file in ZIP.
Comment is string to store as comment with file.

Before call to this method, open archive with OpenZip.
When you finish, call Close to close archive file.
While compressing GetData event is invoked.
If password specified by Password property is wrong an error is invoked.

LastError values

0 No error
-1 General error

BrowseForFolder

Function BrowseForFolder (Title As String, StartupDir As String) As String

Show Browse-For-Folder dialog

Close

Sub Close

Close archive

CreateZip

Sub CreateZip (Filename As String, Comment As String)

Create archive

Arguments

Filename is full path to a ZIP file.

Comment is a global ZIP file comment.

Call this method to create new, empty archive file.

After that you can call AddFile() to add files to archive.

When you finish, call Close to close archive file.

If VolumeSize property is set to ;

- 1 CreateZip method will create PKZIP compatible disk spanning archive on removable disks
- 0 CreateZip method will create a normal ZIP archive
- >0 CreateZip method will create multifile ZIP archive with filesizes up to VolumeSize

LastError values

- 0 No error
- 1 General error

ExtractFile

Sub ExtractFile (Filename As String)

Extract file from archive

Filename is a name of file in archive or wildcards files specification.
To specify more files in one call, use ‘;’ sign as filename delimiter.
Example:

```
FathZIP.ExtractFile("file1.txt ; file2.exe")
```

Call this method to extract one or more files from archive.
Before call to this method, open archive with OpenZip.
When you finish, call Close to close archive file.
While decompressing Status event is invoked.
If password specified by Password property is wrong an error is invoked.
You can use * and ? wildcards in Filename property.
By setting BasePath property you can select where to extract files.

LastError values

- 0 No error
- 1 ZIP not open
- 2 Error opening output file
- 3 File not found in ZIP
- 10 Wrong password
- 11 No password
- 12 Read error
- 13 ZIP file format error

ExtractFile2

Sub ExtractFile2 (SourceFile As String, DestFile As String)

Extract file from archive with new filename

Same as ExtractFile, but with destination filename specified.

ExtractFile2 "data.txt", "newdata.txt"

ExtractToMemory

Sub ExtractToMemory (index As Long)

Extract file from archive to memory

Uncompress data from a file in ZIP into memory.

The DataIn event is triggered to provide you with the data of a file that is being unzipped to memory.

Before call to this method, open archive with OpenZip.

When you finish, call Close to close archive file.

While decompressing DataIn event is invoked.

If password specified by Password property is wrong an error is invoked.

LastError values

- 0 No error
- 1 ZIP not open
- 2 Error opening output file
- 3 File not found in ZIP
- 10 Wrong password
- 11 No password
- 12 Read error
- 13 ZIP file format error

ExtractToMemoryArray

Sub ExtractToMemoryArray (index As Long)

Extract file from archive and return VARIANT(BYTE ARRAY)

Same as ExtractToMemory but, for this method, UncompressedData parameter of DataIn event is of VARIANT (BYTE ARRAY) type.

GzCopy

Sub GzCopy (SrcFile As String, DestFile As String)

Uncompress GZ file

GzCopy "myfile_archive.gz", "myfile.txt"

IsFullVersion

Function IsFullVersion As Long

Returns TRUE if full version of FathZIP is in use

MakeSFX

Sub MakeSFX (exename As String, zipname As String)

Converts ZIP file into self-extracting archive

Arguments

Exename is filename (with path) of output self-extracting program.
ZIPname is a full filename of ZIP archive to convert.

You don't need to open ZIP archive before converting.

If archive contains a file with name 'sfx.ini', it's scanned for options;

Sample sfx.ini would look like this:

```
[Options]
DefaultDir=C:\My Documents ;put TEMP here for Windows temporary
folder
;or PARAM to use command line parameter as default dir.
PromptDir=1 ;1=Ask user to specify the folder, 0=don't ask
AskOverwrite=1 ;1=Ask user before overwriting files, 0=don't ask
Run=myapp.exe
ExtractAll=1 ;1=Yes, 0=extract only program to run
DoneBox=0 ;1=Display done message box, 0=don't display
Title=ZIP self-extractor
Caption=Choose destination folder
Note=Note: This will install...
OK=OK ;OK button caption
Cancel=Cancel ;Cancel button caption
```

LastError values

0 No error
-1 General error

OpenZip

Sub OpenZip (Filename As String)

Open archive for extracting and/or adding files

Arguments

Filename is full path to an existing ZIP file.

Remarks

Call this method to open archive file.
After that you can call ExtractFile() , AddFile or RemoveFile methods.
When you finish, call Close to close archive file.
For spanned archives ChangeDisk event is triggered automatically.

LastError values

- 0 No error
- 1 General error

RemoveFile

Sub RemoveFile (Filename As String)

Remove file from archive

Call this method to remove file from archive.
Before call to this method, open archive with OpenZip.
When you finish, call Close to close archive file.

LastError values

- 0 No error
- 1 Couldn't open ZIP
- 2 Error opening output file
- 3 File not found in ZIP

SetFileComment

Sub SetFileComment (rec As Long, CommentText As String)

Sets comment text for a file in the archive

Arguments

Index is a numeric index of a file in the archive.

Remarks

Before call to this method, open archive with OpenZip.
When you finish, call Close to close archive file.

SetGlobalComment

Sub SetGlobalComment (CommentText As String)

Sets global archive comment text

Before call to this method, open archive with OpenZip.
When you finish, call Close to close archive file.

TestFile

Sub TestFile (index As Long)

Test integrity of the file in archive

If file is OK then return value is 0, else it is negative number and an error is invoked.
If password specified by Password property is wrong an error is invoked.

LastError values

- 0 No error
- 1 ZIP not open
- 2 Error opening output file
- 3 File not found in ZIP
- 10 Wrong password
- 11 No password
- 12 Read error
- 13 ZIP file format error

Events

ChangeDisk

Sub ChangeDisk (Drive As String, DiskNo As Long, Reason As Long, Cancel As Long)

A new disk (or span file) is required while zipping/unzipping spanned archives

The ChangeDisk event is triggered while accessing multidisk archives when there is a need for another disk from disk set.

Drive parameter is a string of type "A:", "C:", etc.

Reason parameter could have one of the following values:

- 1 : a disk needed for reading
- other codes occurs during writing and signalize errors:
- >=0 : number of bytes needed (not enough disk space)
- 2 : the file with the archive name already exists on the disk
- 3 : the disk is probably write - protected
- 4 : couldn't create a file

Set Cancel to True to abort operation.

DataIn

Sub DataIn (Filename As String, Position As Long, UncompressedData, Cancel As Long, EndOfData As Long)

Provide you with the data of a file that is being unzipped to memory.

The DataIn event is triggered to provide you with the data of a file that is being unzipped to memory. The DataIn event is only triggered if you have called ExtractToMemory method.

This event will raise once or more times to complete file decompression.

Filename is name of file that is being decompressed.
Position is a position of a first byte of UncompressedData in file.
Cancel is a flag you can set to TRUE to abort decompression.
EndOfData is a flag indicating is this a last chunk of data.

GetData

Sub GetData (Filename As String, DataToCompress, EndOfData As Long)

Event is triggered so that you can provide the data of a file that is being zipped from memory.

The GetData event is triggered so that you can provide the data of a file that is being zipped from memory. The GetData event is triggered at least once for each file that you have specified with the AddFileFromMemory method.

The GetData event is only triggered if you have called AddFileFromMemory method.

Filename is name of file that is being compressed.

Set EndOfFile to TRUE when there is no more data to compress.

DataToCompress is of data type Variant (String) or Variant (Byte).

OverallStatus

Sub OverallStatus (Filename As String, Progress As Long, Cancel As Long)

Reports compression/decompression overall progress

Filename is name of file that is currently being compressed/decompressed.

Progress is number (percent) between 0 and 100.

Set Cancel to TRUE if you want to abort method.

Status

Sub Status (Filename As String, Progress As Long, Cancel As Long)

Reports compression/decompression progress

Filename is name of file that is being compressed/decompressed.

Progress is number (percent) between 0 and 100.

Set Cancel to TRUE to abort method.

UnzipPreprocessFile

Sub UnzipPreprocessFile (Filename As String, FileDate As Date, FileAttrib As Long, Cancel As Long)

The UnzipPreprocessFile event provides the opportunity to view and change a file's information before it is unzipped.

The UnzipPreprocessFile event provides the opportunity to view and change a file's information before it is unzipped. Filename, date and attributes are modifiable.

This event is raised right before the file is created on a disk while processing ExtractFile method. To cancel extracting this file set Cancel parameter to True.

ZipPreprocessFile

Sub ZipPreprocessFile (Filename As String, FileDate As Date, FileAttrib As Long, FileComment As String, Cancel As Long)

The ZipPreprocessFile event provides the opportunity to view and change a file's information before it is zipped into archive.

The ZipPreprocessFile event provides the opportunity to view and change a file's information before it is zipped into archive. Filename, date, attributes and comment are modifiable.

This event is raised right before the file is created on a disk while processing AddFile method. To cancel zipping this file set Cancel parameter to True.