The Windows Shortcut File Format

as reverse-engineered by
Jesse Hager
jessehager@iname.com
Document Version 1.0

This document assumes that you are familiar with shortcuts and the IShellLink interface. If not, this in probably not the best place to start.

This document is also unofficial, so I don't claim that it is 100% accurate. This information is based solely on the examination of hundreds of shortcut files and comparing them to the documented IShellLink interface. There's still a few things I'm unsure of, namely which time value is which, the contents of the network volume structure and the extra stuff at the end of the file.

If you're writing software under Windows I highly recommend you use the IShellLink interface. For the DOS, Linux, JAVA and other crowds, this is the document you need, 'cause MS isn't gonna give you squat.

Basic File Structure

The file is structured like this:

    File header
    Shell item ID list
        Item 1
        Item 2
        etc..
    File locator info
        Local path
        Network path
    Description string
    Relative path string
    Working directory string
    Command line string
    Icon filename string
    Extra stuff

The File Header

This is of course at the start of the file.

.LNK File Header		
Offset	Size/Type	Contents
0h	1 dword	Always 0000004Ch 'L'
4h	16 bytes	GUID of shortcut files
14h	1 dword	Flags
18h	1 dword	File attributes
1Ch	1 qword	Time 1
24h	1 qword	Time 2
2Ch	1 qword	Time 3
34h	1 dword	File length
38h	1 dword	Icon number
3Ch	1 dword	ShowWnd value
40h	1 dword	Hot key
44h	2 dwords	Unknown, always zero

The first 4 bytes of the file form a long integer that is always set to 4Ch this it the ASCII value for the uppercase letter L. This is used to identify a valid shell link file.

The next 16 bytes is the globally unique identifier GUID of the shell links which is: {00021401-0000-0000-00C0-000000000046} in standard GUID notation or {01h, 14h, 02h, 00h, 00h, 00h, 00h, 00h, C0h, 00h, 00h, 00h, 00h, 00h, 46h} as it is composed of bytes in the file. It appears that in the future, Microsoft may redefine the file format and this will be used to indicate which version to use.
The next item is a long integer which consists of a number of flags. This is important, because it indicates which of the optional parts of the file are present.

The flags		
Bit	Meaning when 1	when 0
0	The shell item id list is present.	The shell item id list is absent.
1	Points to a file or directory.	Points to something else.
2	Has a description string.	No description string.
3	Has a relative path string.	No relative path.
4	Has a working directory.	No working directory.
5	Has command line arguments.	No command line arguments.
6	Has a custom icon.	Has the default icon.

The next item is a long integer that contains file attributes of the target file. If the target is not a file (see flags bit 1), then this is set to zero. The resolver uses these when the link is broken to match the link with the correct target.

File Attributes	
Bit	Meaning when set
0	Target is read only.
1	Target is hidden.
2	Target is a system file.
3	Target is a volume label. (Not possible)
4	Target is a directory.
5	Target has been modified since last backup. (archive)
6	Target is encrypted (NTFS EFS)
7	Target is Normal??
8	Target is temporary.
9	Target is a sparse file.
10	Target has reparse point data.
11	Target is compressed.
12	Target is offline.

The next three items are 64 bit integers that specifiy the various time information for the file.

Creation time
Modification time
Last access time

The next item is a long integer which contains the length of the target file.

If the file has a custom icon (see flags bit 6), then this long integer indicates the index of the icon to use. Otherwise it is zero.

The next long integer specifies the ShowWnd value to pass to the target application when starting it. For your convenience, the values are reproduced below. It is unlikely, that most of these values are valid. Only values 1, 2 and 3 are permitted in the shortcut property page.

SW_HIDE	0	Cool...
SW_NORMAL	1	
SW_SHOWMINIMIZED	2	
SW_SHOWMAXIMIZED	3	
SW_SHOWNOACTIVATE	4	
SW_SHOW	5	
SW_MINIMIZE	6	
SW_SHOWMINNOACTIVE	7	
SW_SHOWNA	8	
SW_RESTORE	9	
SW_SHOWDEFAULT	10	

The next long integer specifies the hotkey assigned to the shortcut.

The last two long integers are always zero. They are probably reserved for future use.

The Shell Item Id List.

This item is only present if bit 0 is set in the flags word of the header.

An entire book could be written on the contents of this item. Essentially it indicates how to get from the desktop to the specified item. The actual contents are highly variable. The following are the only constant items about the list.

The first unsigned short integer indicates the total length of the list so it can be skipped easily.

Inside the list, each item begins with an unsigned short integer that indicates the length of the item. The length includes the size of the length value.

The last item is length 0.

Lookup ITEMIDLIST in most any Win32 documentation for more info on this item.

File Location Info

This item is always present, but if bit 1 is not set in the flags value, then the length of this structure will be zero. The following table shows the structure of the header of this item.

File Location Info		
Offset	Size	Contents
0h	1 dword	This is the total length of this structure and all following data
4h	1 dword	This is a pointer to first offset after this structure. 1Ch
8h	1 dword	Flags
Ch	1 dword	Offset of local volume info
10h	1 dword	Offset of base pathname on local system
14h	1 dword	Offset of network volume info
18h	1 dword	Offset of remaining pathname
Notes: The first length value includes all the assorted pathnames and other data structures. All offsets are relative to the start of this structure.		

The first long integer indicates the size of the file location info.

The next long integer is the offset at which the basic file info structure ends. Should be 1Ch under normal conditions.

The next long integer is the flags that indicate which types of volumes the file is available on.

Volume flags	
Bit	Meaning
0	Available on a local volume
1	Available on a network share

The next long integer is the offset to the local volume table. (See below)
   (Warning: Random garbage when bit 0 is clear in volume flags)

The next long integer is the offset to the base path on the local volume.
   (Warning: Random garbage when bit 0 is clear in volume flags)

The next long integer is the offset to the network volume table. (See below)
   (Warning: Random garbage when bit 1 is clear in volume flags)

The next long integer is the offset to the final part of the pathname.

To find the filename of the file on the local volume, combine the base path string and the final path string.

To find the filename of the file on the network, combine the share name in the network volume table with the final path string.

The local volume table		
Offset	Size	Contents
0h	1 dword	Length of this structure.
4h	1 dword	Type of volume
8h	1 dword	Volume serial number
Ch	1 dword	Offset of the volume name (Always 10h)
10h	ASCIZ	Volume label

The first long integer in the local volume table is the length of the structure including the volume label string.

The next long integer is the type of volume.

0	Unknown
1	No root directory
2	Removable (Floppy, Zip, etc..)
3	Fixed (Hard disk)
4	Remote (Network drive)
5	CD-ROM
6	Ram drive (Shortcuts to stuff on a ram drive, now that's smart...)

The next long integer is the volume serial number.

The next long integer is the offset of the volume label within the structure. Always 10h under normal conditions.

The network volume table		
Offset	Size	Contents
0h	1 dword	Length of this structure
4h	1 dword	Unknown, always 2h?
8h	1 dword	Offset of network share name (Always 14h)
Ch	1 dword	Unknown, always zero?
10h	1 dword	Unknown, always 20000h?
14h	ASCIZ	Network share name
Note 1: The above unknown values are the same for a printer or file share. Note 2: The above values are for Microsoft Networks, I don't have a NetWare server to test.		

The first long integer is the length of the structure including the length of the network share name.

The next long integer is unknown, it seems to always be 2h on Microsoft Networks.

The next long integer is the offset to the share name within the structure.

The next two long integers are unknown.

The share name specifies the share name that the item is available under.

Description string

If bit 2 is set in the flags value in the header, then this string is present.

The first unsigned short int value indicates the length of the string. Following the length value is a string of ASCII characters. It is a description of the item.

Relative path string

If bit 3 is set in the flags value in the header, then this string is present.

The first unsigned short int value indicates the length of the string. Following the length value is a string of ASCII characters. It is a relative path to the target.

Working directory

If bit 4 is set in the flags value in the header, then this string is present.

The first unsigned short int value indicates the length of the string. Following the length value is a string of ASCII characters. It is the working directory as specified in the shortcut properties.

Command line string

If bit 5 is set in the flags value in the header, then this string is present.

The first unsigned short int value indicates the length of the string. Following the length value is a string of ASCII characters. The command line string includes everything except the program name.

Icon filename string

If bit 6 is set in the flags value in the header, then this string is present.

The first unsigned short int value indicates the length of the string. Following the length value is a string of ASCII characters. This the name of the file containing the icon.

Extra stuff

The last item in the file is usually a long integer with the value zero. In rare cases, this long integer seems to be the length of some unknown structure that follows.

The only values I've ever seen in here are:

1 dword	10h	Length of following data
1 dword	A0000005h	?
1 dword	1Ah	?
1 dword	6Ch	?
1 dword	0	?

Another possible arrangement is:

1 dword	10h	Length of first structure
3 dwords	x	Remainder of first structure
1 dword	0	Length of next structure

Disassembly of a hypothetical shortcut file

Offset	Bytes	Contents
Header		
0000	4C 00 00 00	'L' Magic value
0004	01 04 02 00	GUID of shortcut files
	00 00 00 00	
	C0 00 00 00	
	00 00 00 46	
0014	3F 00 00 00	Flags
		Has item id list
		Target is a file
		Has description string
		Has relative pathname
		Has a working directory
		Has a custom icon
0018	20 00 00 00	File attibutes
		Archive
001C	C0 0E 82 D5	Time 1
	C1 20 BE 01	
0024	00 08 BF 46	Time 2
	D5 20 BE 01	
002C	00 47 AA EC	Time 3
	EC 15 BE 01	
0034	A0 86 00 00	File length is 34464 bytes. 86A0h
0038	05 00 00 00	Icon number 5
003C	01 00 00 00	Normal window
0040	46 06 00 00	Ctrl-Alt-F hotkey
0044	00 00 00 00	Always zero, unknown/reserved
0048	00 00 00 00	Always zero, unknown/reserved
Item Id List		
004C	2A 00	Size of item id list
First item		
004E	28 00	Length of first item
0050	32 00	???
0052	A0 86 00 00	File length
0056	76 25 71 3E	???
005A	20 00	File attributes?
005C	62 65 73 74 5F 37	"best_773.mid" Long name
	37 33 2E 6D 69 64	
	00	Null terminator
0069	42 45 53 54 5F 37	"BEST_773.MID" Short name
	37 33 2E 4D 49 44	
	00	Null terminator
Last item		
0076	00 00	Zero length value

```
Offset      Bytes                  Contents
```

File location info

```
0078      74 00 00 00            Structure length
007C      1C 00 00 00            Offset past last item in structure
0080      03 00 00 00            Flags
                                       Local volume
                                       Network volume
0084      1C 00 00 00            Offset of local volume table
0088      34 00 00 00            Offset of local path string
008C      40 00 00 00            Offset of network volume table
0090      5F 00 00 00            Offset of final path string
```

Local volume table

```
0094      18 00 00 00            Length of local volume table
0098      03 00 00 00            Fixed disk
009C      D0 07 33 3A            Volume serial number 3A33-07D0
00A0      10 00 00 00            Offset to volume label
00A4      44 52 49 56 45 20      "DRIVE C",0
          43 00
00AC      43 3A 5C 57 49 4E      "C:\WINDOWS\" local path string
          44 4F 57 53 5C 00
```

Network volume table

```
00B8      1F 00 00 00            Length of network volume table
00BC      02 00 00 00            ???
00C0      14 00 00 00            Offset of share name
00C4      00 00 00 00            ???
00C8      00 00 02 00            ???
00CC      5C 5C 4A 45 53 53      "\\JESSE\WD",0      Share name
          45 5C 57 44 00
00D7      44 65 73 6B 74 6F      "Desktop\best_773.mid",0
          70 5C 62 65 73 74      Final path name
          5F 37 37 33 2E 6D
          69 64 00
```

Description string

```
00EC      12 00                  Length of string
00EE      42 65 73 74 20 37      "Best 773 midi file"
          37 33 20 6D 69 64
          69 20 66 69 6C 65
```

Relative path

```
0100      0E 00                  Length of string
0102      2E 5C 62 65 73 74      ".\best_773.mid"
          5F 37 37 33 2E 6D
          69 64
```

Working directory

```
0114      12 00                  Length of string
0116      43 3A 5C 57 49 4E      "C:\WINDOWS\Desktop"
          44 4F 57 53 5C 44
          65 73 6B 74 6F 70
```

```
Offset     Bytes                  Contents
```
Command line arguments
```
0128       06 00
012A       2F 63 6C 6F 73 65      "/close"
```
Icon file
```
0130       16 00                  Length of string
0132       43 3A 5C 57 49 4E      "C:\WINDOWS\Mplayer.exe"
           44 4F 57 53 5C 4D
           70 6C 61 79 65 72
           2E 65 78 65
```
Ending stuff
```
0148       00 00 00 00            Length 0 - no more stuff
```


The target is located at:
        C:\WINDOWS\Desktop\best_773.mid

The windows directory is shared as:
        \\JESSE\WD