

New Technical Notes

Macintosh



Developer Support

Scrapbook File Format Toolbox

Written by: Bill Guschwan and Chris Beraducci

May 1993

This Technical Note discusses the format of the System 7.1 Scrapbook file format. The format may change in the future so be warned.

Topics

- The format of the 'SMAP' resource
-

Introduction

This Technical Note discusses the format of the Scrapbook file. Previously the format was undocumented. Developers, however, have consistently asked for documentation of the format, so we are providing it with a strong word of warning. The format may change in the future, and your application risks compatibility if you rely on it.

If you have decided to rely on the Scrapbook format, you should always code for a specific version of the Scrapbook file. The Scrapbook file will always have a 'vers' resource, which you can check.

Scrapbook Brief Overview

The Scrapbook consists of a Scrapbook application and a Scrapbook file. The Scrapbook file is stored at the root of the System Folder. The Scrapbook application is usually stored in the Apple Menu Items folder.

Scrapbook File Format Overview

The resource fork of the Scrapbook file contains two permanent resources ('SMAP' and 'vers'), and up to 255 scrap items. The data fork is empty. Thus, any empty Scrapbook file will contain both a 'vers' resource and an 'SMAP' resource. Each scrap item will have a variable number of associated resources. It may also have a dependent alias, as discussed below.

Check the 'vers' Resource

The Scrapbook file is part of the Scrapbook application. Because the Scrapbook application is part of system software, the Scrapbook file format is highly subject to change. If you choose to open the file, you will need to identify the Scrapbook file format version by checking the 'vers' resource in the Scrapbook file. Your access of the Scrapbook file should be subject to the version of the file that your application understands. If you do not understand the version, then your application should not access the file. Your application should *never* have Scrapbook File format access as a requirement.

Format of the 'SMAP' Resource

The 'SMAP' resource is the key to the Scrapbook file format. The 'SMAP' resource is a resource of 255 bytes. It is a three-dimensional table of information. The 'SMAP' keeps track of the resource IDs for Scrapbook items on one axis, and it keeps track of the order of the Scrapbook items on the other.

Scrapbook Items

The Scrapbook application works with Scrapbook items, which are groupings of related resources. Many objects associated with the Macintosh desktop contain more than one resource. An icon, for example, has about six resources associated with it. The Scrapbook, however, needs to treat these groupings as one item. To the end user, he or she is merely pasting in an icon and removing it.

A Scrapbook item is uniquely identified to the Scrapbook by a resource ID. All resources with the same resource ID belong to the same Scrapbook item. Thus, an icon's resources would be considered a Scrapbook item.

Generating Unique Resource IDs

When you paste a Scrapbook item in to the Scrapbook, the Scrapbook needs to generate a unique resource ID for that item. And because Scrapbook items can be cut and pasted at will, this method needs some way of recovering resource IDs that are no longer being used.

The location of each of the 255 bytes in the 'SMAP' resource represents a unique ID. For example, the first byte in the 'SMAP' is associated with a resource ID of -32768. The second byte is associated with a resource ID of -32767. This continues sequentially until the Scrapbook limit of 255 items (a resource ID of -32512) is reached.

A used resource ID is represented by a nonzero value at that byte's position. A zero indicates an unused resource ID. When the Scrapbook needs a unique ID, it sequentially scans the

Scrapbook file for a zero byte, calculates the resource ID, and uses it for the new Scrapbook item.

On the other hand, when a Scrapbook item is cut, the application first cuts all the resources of the corresponding resource ID. It then finds the Scrapbook item's resource ID in the 'SMAP' resource, and sets its value to zero.

Maintaining the Ordering of Scrapbook Items

The 'SMAP' file also maintains the order of Scrapbook items in the Scrapbook. The nonzero value at each location in an 'SMAP' is the location in the Scrapbook of the Scrapbook item. The value of the byte corresponds to the entry number in the Scrapbook.

Example

For example, in the 'SMAP' resource of your Scrapbook data file you may find the following (looking with ResEdit at the 'SMAP' resource):

```
0A07 0509 0801 0602
0403 0000 0000 0000
```

Thus, if we take the first byte, the tenth Scrapbook item is identified by the resource ID-32768.

The information further translated:

Res ID	Byte offset	Entry in Scrapbook	Entry in Scrapbook (Hex\$)	Type
-32768	00	10	A	moov
-32767	01	7	7	PICT
-32766	02	5	5	PICT
-32765	03	9	9	PICT
-32764	04	8	8	PICT
-32763	05	1	1	PICT
-32762	06	6	6	snd
-32761	07	2	2	PICT (QT)
-32760	08	4	4	PICT (QT)
-32759	09	9	9	PICT (QT)

Thus, the snd resource is the sixth Scrapbook item in the Scrapbook. Its resource ID is $-32768 + 5 = -32762$. The fourth item in the Scrapbook is a QuickTime picture with resource ID -32760.

Scrapbook and QuickTime

When you paste a QuickTime movie into the Scrapbook, you will copy only the movie resource. Remember, even if you cut a movie from a self-contained movie, only the movie resource will be cut. When you paste the movie into the Scrapbook, the movie gets pasted into the Scrapbook's file and the file is now a dependent file. The `FinderInfo` of the Scrapbook file will now have its shared bit set, and a dependent alias will be put in the resource fork of the Scrapbook file. When the movie is cut from the Scrapbook, its shared bit should be reset and the dependent alias needs to be removed. Unfortunately, this does not occur with all versions of the Scrapbook because they do not support QuickTime.