

MiniMacRezMgr

By Steve Stephenson

This collection of routines function as a read-only GS-style Resource Manager for Macintosh files.

It is patterned after the GS Resource Manager. The parameters are the same (except as noted), the tool function numbers are retained as the low byte of the message numbers, and the low byte of the error numbers are retained.

No attempt (insane!) is made to write to the Macintosh disk; therefore, several functions that are present in the GS Resource Manager are not needed here.

For this first version, only one caller may log in at a time. And the caller may only have one file open at a time. So there is no need for Get/SetCurResourceApp or Get/SetCurResourceFile.

General Notes

Integer and long integer data on the Macintosh is in a reversed order from that in the GS.

Example:	\$12345678
GS:	78 56 34 12
Macintosh:	12 34 56 78

The data returned to the caller by this program is left in its original Macintosh order.

On the GS, resource types are 2 bytes in length and IDs are 4 bytes. On the Macintosh, these are reversed; types are 4 bytes and IDs are 2 bytes.

On the GS, the number of types in a file is returned in a long int. On the Macintosh, the number of both types and resources is restricted to a word.

Macintosh resource attributes have no meaningful correlation on the GS and are ignored. All loaded resources are returned in unlocked, unrestricted handles.

Error Codes

BB02 mrBadFormat
BB06 mrNotFound
BB07 mrFileNotFound
BB08 mrBadAppID
BB0A mrIndexRange
BB0F mrInvalidShutdown
BB10 mrNameNotFound
BB12 mrDupStartup
BBFF mrBusy
0121 srqDuplicateName

How to use it

Load the support routine using BFLoadSupport. The request procedure name is:

"Babelfish Support~Synesis Systems~MiniMacRezMgr~"

The first call (or message/request) sent to the RezMgr must be `mrStartup`. This establishes you as the user and blocks all other callers until you call `mrShutdown`.

To open a file, pass its path name to `mrOpenResourceFile`. You should have already screened this file to make sure it is a Macintosh resource file. Perhaps with a `SFGetFile` filter proc that checks for all of the following: 1) Filetype is zero, 2) Auxtype is zero, 3) File is extended, 4) Resource EOF is non-zero.

While Macintosh resource types are documented, it is very unlikely you will know the resource IDs or resource names ahead of time. You will have to dig them out of the map.

To find all resource IDs of a particular type, start by calling `mrCountResources`. With the absolute count that is returned, you could loop while calling `mrGetIndResource` and incrementing the loop counter from one to the total number. Alternately, you could call `mrGetIndResource` with an increasing index until you get the out of range error.

mrInit

Resource Type: `rCodeResource` (\$8017) Resource ID: \$00000001 Attributes: locked, convert

Description

This function is called with a JSL. It installs the request proc command handler and returns with a RTL. The accumulator and carry are returned unchanged from the `_AcceptRequests` call. The only error would result from attempting to install the handler more than once.

Parameters

The stack is not affected by this call. There are no input or output parameter.

Errors

`srqDuplicateName` This message handler already installed

Notes

The format of the support file's request name is "Babelfish Support~Synesis Systems~MiniMacRezMgr~".

mrStartUp (\$B002)

Description

Logs in the caller as the “owner” of the RezMgr.

Parameters

dataIn — Pointer to this structure:

\$00	UserID	Word—The memory ID of the caller
------	--------	----------------------------------

dataOut — Pointer to this structure:

\$00	recvCount	Word—Number of times the request was received
\$02	mrResult	Word—Error code

Errors

mrDupStartup	RezMgr already started for this user
mrBusy	another user already has RezMgr started

mrShutdown (\$B003)

Description

This code shuts down the RezMgr. It also closes the resource file if still open.

Parameters

dataIn — Pointer to this structure:

\$00	UserID	Word—The memory ID of the caller
------	--------	----------------------------------

dataOut — Pointer to this structure:

\$00	recvCount	Word—Number of times the request was received
\$02	mrResult	Word—Error code

Errors

mrInvalidShutdown	RezMgr not started
mrBadAppID	UserID invalid

mrGetVersion (\$B004)

Description

Returns the version number of RezMgr in ToolSet format.

Parameters

dataIn — NIL

dataOut — Pointer to this structure:

\$00	recvCount	Word-Number of times the request was received
\$02	mrResult	Word-Error code
\$04	version	Word-Version number

Errors

none

mrOpenResourceFile (\$B00A)

Description

Opens the specified file's resource fork read-only. It is up to the caller to ensure that this path leads to a Macintosh file that has a resource fork containing Macintosh resources as only minimal checking is done.

Parameters

dataIn — Pointer to this structure:

\$00	pathPtr	Long-Pointer to C1InputString containing the path name
------	---------	--

dataOut — Pointer to this structure:

\$00	recvCount	Word-Number of times the request was received
\$02	mrResult	Word-Error code
\$04	fileID	Word-ID number for this file

Errors

mrBadFormat not a Macintosh resource file
mrBusy another file is already open
GS/OS and Memory Mgr errors returned unchanged.

Notes

Save the fileID number as it will be needed to close the file.

mrCloseResourceFile (\$B00B)

Description

Closes the Macintosh resource file, disposes of all resource handles in the map and the map itself.

Parameters

dataIn — Pointer to this structure:

\$00	fileID	Word—The file ID of the open file
------	--------	-----------------------------------

dataOut — Pointer to this structure:

\$00	recvCount	Word—Number of times the request was received
\$02	mrResult	Word—Error code

Errors

mrFileNotFound fileID is invalid

mrLoadResource (\$B00E)

Description

Returns a handle containing the data of this resource type and ID.

Parameters

dataIn — Pointer to this structure:

\$00	— type —	Long—Resource type
\$04	ID	Word—Resource ID

dataOut — Pointer to this structure:

\$00	recvCount	Word—Number of times the request was received
\$02	mrResult	Word—Error code
\$04	— handle —	Long—Data handle

Errors

mrNotFound no resource found matching this type and ID
GS/OS and Memory Mgr errors returned unchanged.

Notes

No manipulation of the data is done. It is completely up to the caller to understand the significance of the data.

mrReleaseResource (\$B017)

Description

Sets the purge level (0-3) of the specified resource, if loaded (returns no error if not). If the purge level is negative the resource is immediately disposed of.

Parameters

dataIn — Pointer to this structure:

\$00	—	type	—	Long-Resource type
\$04		ID		Word-Resource ID
\$06		purgeLevel		Word-purge level

dataOut — Pointer to this structure:

\$00		recvCount		Word-Number of times the request was received
\$02		mrResult		Word-Error code

Errors

mrNotFound no resource found matching this type and ID

Notes

The position of `purgeLevel` in the input parameters is different from the GS Resource Mgr.

mrDetachResource (\$B018)

Description

Removes the handle of the resource from the map. The handle then belongs to the caller for management.

Parameters

dataIn — Pointer to this structure:

\$00	—	type	—	Long-Resource type
\$04		ID		Word-Resource ID

dataOut — Pointer to this structure:

\$00		recvCount		Word-Number of times the request was received
\$02		mrResult		Word-Error code

Errors

mrNotFound no resource found matching this type and ID

mrGetResourceSize (\$B01D)

Description

Returns the length of the resource's data on disk.

Parameters

dataIn — Pointer to this structure:

\$00	—	type	—	Long-Resource type
\$04		ID		Word-Resource ID

dataOut — Pointer to this structure:

\$00		recvCount		Word-Number of times the request was received
\$02		mrResult		Word-Error code
\$04	—	size	—	Long-Length of the resource's data

Errors

mrNotFound no resource found matching this type and ID
GS/OS and Memory Mgr errors returned unchanged.

Notes

The length of a resource is not maintained within the map, therefore the disk must be accessed to determine it.

mrCountTypes (\$B020)

Description

Returns the number of resource types in the file.

Parameters

dataIn — NIL

dataOut — Pointer to this structure:

\$00		recvCount		Word-Number of times the request was received
\$02		mrResult		Word-Error code
\$04		count		Word-Number of resource types

Errors

none

Notes

The count returned is one-relative (even though it is zero-relative on disk).

mrGetIndType (\$B021)

Description

Returns the resource type found at the indexed position in the map.

Parameters

dataIn — Pointer to this structure:

\$00	index	Word—The relative number type to find
------	-------	---------------------------------------

dataOut — Pointer to this structure:

\$00	recvCount	Word—Number of times the request was received
\$02	mrResult	Word—Error code
\$04	type	Long—Resource type

Errors

mrIndexRange index out of range (either zero or past last type)

Notes

The resource type is not reversed; it is returned in Macintosh order.

mrCountResources (\$B022)

Description

Returns the number of resources of this type. If the type is not found, the returned count will be zero.

Parameters

dataIn — Pointer to this structure:

\$00	type	Long—Resource type
------	------	--------------------

dataOut — Pointer to this structure:

\$00	recvCount	Word—Number of times the request was received
\$02	mrResult	Word—Error code
\$04	count	Word—Number of resource types

Errors

none

Notes

The size of `count` is a word (as on the Macintosh), rather than a long (as on the GS). The `count` is one-relative (even though it is zero-relative on disk).

mrGetIndResource (\$B023)

Description

Returns the resource ID found at the indexed position in the map for this type.

Parameters

dataIn — Pointer to this structure:

\$00	—	type	—	Long-Resource type
\$04		index		Word-The relative number resource to find

dataOut — Pointer to this structure:

\$00		recvCount		Word-Number of times the request was received
\$02		mrResult		Word-Error code
\$04		ID		Word-Resource ID

Errors

mrIndexRange index out of range (either zero or past last resource)

Notes

The size of the ID is a word (as on the Macintosh), rather than a long (as on the GS). The resource ID is not reversed; it is returned in Macintosh order.

mrFindNamedResource (\$B02A)

Description

Returns the ID of this resource type and name. This finds a resource within the specified type with the input name. The name must match exactly (case sensitive). It returns the same error if either the type or the name are not found.

Parameters

dataIn — Pointer to this structure:

\$00	—	type	—	Long-Resource type
\$04	—	namePtr	—	Long-Pointer to name

dataOut — Pointer to this structure:

\$00	—	recvCount	—	Word-Number of times the request was received
\$02	—	mrResult	—	Word-Error code
\$04	—	ID	—	Word-Resource ID

Errors

mrNameNotFound no resource found matching this type and name
GS/OS and Memory Mgr errors returned unchanged.

Notes

The size of the ID is a word (as on the Macintosh), rather than a long (as on the GS). The resource ID is not reversed; it is returned in Macintosh order.

As RezMgr only allows a single file, the GS Resource Manager's fileNumPtr is not used.

mrGetResourceName (\$B02B)

Description

Returns the name of this resource type and ID. Copies the PString into the caller's buffer. No check is made for length. The caller should allow for string lengths up to 255 characters. This call returns the same error if either the type or ID is not found or is not named.

Parameters

dataIn — Pointer to this structure:

\$00	—	type	—	Long-Resource type
\$04		ID		Word-Resource ID
\$06	—	namePtr	—	Long-Pointer to name buffer

dataOut — Pointer to this structure:

\$00		recvCount		Word-Number of times the request was received
\$02		mrResult		Word-Error code

Errors

mrNameNotFound no named resource found matching this type and ID

mrLoadNamedResource (\$B02C)

Description

Returns a handle containing the data of this resource type and name. This finds a resource within the specified type with the input name. The name must match exactly (case sensitive). It returns the same error if either the type or the name are not found.

Parameters

dataIn — Pointer to this structure:

\$00	—	type	—	Long-Resource type
\$04	—	namePtr	—	Long-Pointer to name

dataOut — Pointer to this structure:

\$00	—	recvCount	—	Word-Number of times the request was received
\$02	—	mrResult	—	Word-Error code
\$04	—	handle	—	Long-Data handle

Errors

mrNameNotFound no resource found matching this type and name
GS/OS and Memory Mgr errors returned unchanged.

Notes

No manipulation of the data is done. It is completely up to the caller to understand the significance of the data.