$MENU32 & $MN32 Released in GSM SP-1, SP-2 & SP-3

1. Introduction

This document provides some technical information regarding the 32-bit Menu Handler ($MENU32) and the 32-bit Menu Maintenance utility ($MN32) distributed with GSM Service Pack 1 (GSM-SP1).

In this document the following terms will be used:

16-bit Menu **Handler**	$MH (and related overlays)
32-bit Menu **Handler**	$MENU32 (and the BO$MEN 32-bit DLM)
16-bit Menu Maintenance **utility**	MN (and related overlays)
32-bit Menu Maintenance **utility**	$MN32 (and various 32-bit DLM's and programs)
16-bit Menu **file**	Any menu file (e.g. $$MENUS) created and maintained using 16-bit Menu Maintenance utility (i.e. MN);
32-bit Menu **file**	Any menu file (e.g. $$MENUS) created and maintained using 32-bit Menu Maintenance utility (i.e. $MN32);

2. 16-bit Menu handler ($MH) options

The 16-bit Menu Handler allows both 32-bit programs and frames to be run (in addition to 16-bit frames and programs). This table summarizes the "well known" options:

16-bit Menu Handler ($MH) with 16-bit Menu file				
Menu line type	**Run in 32-bit flag**	**Program/frame type**	**Result**	**Notes**
S	Not available	16-bit program	OK	1
S	Not available	32-bit program	OK	1
F	N	16-bit frame	OK	2
F	N	32-bit frame	ILL PAGE AT D5CC	3
F	Y	16-bit frame	STOP 25015	4
F	Y	32-bit frame	OK	3

E	N	16-bit frame	OK	2
E	N	32-bit frame	ILL PAGE AT D5CC	3
E	Y	16-bit frame	STOP 25015	4
E	Y	32-bit frame	OK	3
Others	Not available	Not applicable	See $MH doc'n	5

Note-1 It is well-known that both 16-bit programs and 32-bit programs (i.e. applications that do not require open Speedbase databases) can invoked from type "S" menu lines without any special precautions.

Note-2 It is well-known that 16-bit frames (i.e. Speedbase applications that require open Speedbase database(s)) can invoked from type "F" and "E" menu lines.

Note-3 In order to run a 32-bit frame (i.e. a Speedbase application that requires an open Speedbase database(s)) from a type "F" or "E" menu line the "Run frame in 32-bit mode" flag must be set to "Y". A small 32-bit assist program, $MH32, is used to load the 32-bit frame.

A Menu file upgrade program, MN32M, is available upon request to set the "Run frame in 32-bit mode" flags on ALL type "F" and "E" menu lines to "Y".

Note-4 If the "Run frame in 32-bit mode" flag is inadvertently set to "Y" for a 16-bit frame a STOP 25015 will result.

Note-5 This document only considers type "S", "F" and "E" menu lines. All other menu line types (e.g. "M", "N", "A" etc.) are unaffected by replacing $MH by $MENU32; or by using $MN32 instead of MN.

2. $MENU32 Overview

An evaluation version of $MENU32 was provided with the initial release of GSM V8.1l. The version of $MENU32 included in GSM-SP1 represents the first fully supported version of $MENU32.

$MENU32 provides a 32-bit "super-set" equivalent of the 16-bit $MH. In addition to significant extra functionality (see below) $MENU32 integrates directly with the Global Application Explorer GX to provide the GX-compatible "button style" menus. This integration with GX is usually the most compelling reason to switch from 16-bit $MH to 32-bit $MENU32.

Important Note: The "GX button" style menu is automatically enabled when $MENU32 is running on GX regardless of the actual Menu Display style (i.e. GSM, SAA and Pop-Menu style menus are all displayed in GX button style when $MENU32 is running on GX).

When the version of $MENU32 released with GSM SP-1 or SP-2 is running with a 16-bit Menu file the above table becomes:

32-bit Menu Handler ($MENU32) with 16-bit Menu file (GSM SP-1 and SP2)				
Menu line type	**Run in 32-bit flag**	**Program/frame type**	**Result**	**Notes**
S	Not available	16-bit program	OK	1
S	Not available	32-bit program	OK	1
F	N	16-bit frame	STOP 25015	2
F	N	32-bit frame	OK	3
F	Y	16-bit frame	STOP 25015	2
F	Y	32-bit frame	OK	3
E	N	16-bit frame	STOP 25015	2
E	N	32-bit frame	OK	3
E	Y	16-bit frame	STOP 25015	2
E	Y	32-bit frame	OK	3
Others	Not available	Not applicable	See $MH doc'n	

Note-1 Both 16-bit programs and 32-bit programs (i.e. applications that do not require open Speedbase databases) can invoked from type "S" menu lines by $MENU32 without any special precautions.

Note-2 **For GSM SP-1 and SP-2, the combination of the 32-bit Menu Handler, $MENU32, and a 16-bit Menu file cannot be used to invoke a 16-bit Speedbase frame regardless of the setting of the "Run frame in 32-bit mode" flag**.

Note-3 The setting of the "Run frame in 32-bit mode" flag is ignored by $MENU32.

For GSM SP-3 $MENU32 has been improved so that the above table becomes:

32-bit Menu Handler ($MENU32) with 16-bit Menu file (GSM SP-3)				
Menu line type	**Run in 32-bit flag**	**Program/frame type**	**Result**	**Notes**
S	Not available	16-bit program	OK	1
S	Not available	32-bit program	OK	1
F	N	16-bit frame	OK	2
F	N	32-bit frame	STOP 25015	2
F	Y	16-bit frame	NO BASE @ 3DEC	2

F	Y	32-bit frame	OK	2
E	N	16-bit frame	OK	2
E	N	32-bit frame	STOP 25015	2
E	Y	16-bit frame	NO BASE @ 3DEC	2
E	Y	32-bit frame	OK	2
Others	Not available	Not applicable	See $MH doc'n	

Note-1 Both 16-bit programs and 32-bit programs (i.e. applications that do not require open Speedbase databases) can be invoked from type "S" menu lines by $MENU32 without any special precautions.

Note-2 **For GSM SP-3 the combination of the 32-bit Menu Handler, $MENU32, and a 16-bit Menu file CAN be used to invoke a 16-bit Speedbase frame. The setting of the "Run frame in 32-bit mode" flag is now correctly recognised**.

3. $MN32 and 32-bit Menu files

The $MN32 32-bit Menu Maintenance utility can be used to convert a 16-bit Menu file to a 32-bit Menu file. Such a conversion provides the following immediate benefits:

- The maximum number of unit assignments per menu line has been increased from 8 to 48;

- The maximum number of open databases per menu line has been increased from 8 (16-bit type "E" menu line) to 32;

- A new option allows a context-sensitive Windows help file to be associated with each menu line. This option is only recognised when the menu is displayed as a GX "button-style" menu. The context sensitive help is selected by a combination of the "GX Help file" option which allows the (2 character) name of a compiled HTML help-file to be specified (e.g. CL.HTM) and an "index" option which specifies a 12-character, free-format tag into the help file ;

- The type "E" menu line type has been de-conceived. Type "E" menu lines were only added to 16-bit Menu files to allow an increase in the number of databases from 4 to 8. Since 32-bit Menu files allow a maximum of 32 databases, this difference has disappeared. Both 16-bit type "F" and type "E" menu lines are converted to 32-bit type "F" menu lines;

- A new "Run frame in 16-bit mode" flag is available in 32-bit Menu files (see below for further details); **Important Note**: The setting of the original "Run in 32-bit mode" flag in the 16-bit Menu file is ignored when converting to a 32-bit Menu file. The default setting of the "Run in 16-bit mode" flag in a freshly converted 32-bit Menu file is "N" (regardless of the setting of the "Run in 32-bit mode" flag in the original 16-bit Menu file);

However, the following ramifications must be heeded:

- **The 16-bit Menu Handler, $MH, cannot be used with a 32-bit Menu file**. The following error message will appear if an attempt is made to use the 16-bit Menu Handler with a 32-bit Menu file:

  MENU FILE VERSION IS TOO HIGH - RUN $MENU32

  Note that some early versions of $MH V8.1 may not display this explicit warning message but may display a garbled menu;

- **The 16-bit Menu Maintenance utility, MN, will not recognise a 32-bit Menu file**. The following error message will appear if an attempt is made to use 16-bit Menu Maintenance with a 32-bit Menu file:

  INVALID MENU FILE

- **A 32-bit Menu file is significantly larger than the equivalent 16-bit Menu file.** Although the exact size increase is difficult to predict precisely, a 5-fold size increase is typical.

- The version of $MN32 released with GSM SP-1 and GSM SP-2 did not support an option to print a 32-bit menu file. This shortcoming has been addressed with the version of $MN32 released with GSM SP-3.

WHEN $MN32 IS USED TO AMEND AN EXISTING MENU FILE THE RESULT IS ALWAYS A 32-BIT MENU FILE. THE VERSION OF $MN32 RELEASED WITH GSM SP-1 DOES NOT OFFER A CONVERSION OPTION. THIS ISSUE HAS BEEN FIXED WITH GSM SP-2 (I.E. THE VERSION OF $MN32 RELEASED WITH GSM SP-2 WARNS THAT THE RESULANT 32-BIT MENU WILL ONLY BE USEABLE WITH $MENU32).

ONCE A MENU FILE HAS BEEN CONVERTED TO 32-BIT IT CANNOT BE CONVERTED BACK TO A 16-BIT MENU FILE.

ONCE A MENU FILE HAS BEEN CONVERTED TO 32-BIT IT CAN ONLY BE USED WITH $MENU32. A 32-BIT MENU FILE CANNOT BE USED WITH $MH.

Important Note-1: The version of $MN32 that was released in some pre-release versions of GSM Service Pack-1 contained a bug that created a corrupt 32-bit Menu file if the original 16-bit Menu file contained one, or more, Access Rights. This problem has been fixed for the "official" version of GSM Service Pack 1 released on the Oct-2000 Revision 1 Global Product Set CD. The "fixed" version of $MN32, in library P.$CMLB2, is dated 11/10/2000.

Important Note-2: A further, more obscure, menu corruption problem was fixed by the version of $MN32 included in GSM SP-2.

4. $MH and 32-bit Menu files

As explained above the 16-bit Menu Handler cannot be used with a 32-bit Menu file.

5. $MENU32 and 32-bit Menu files

The combination of $MENU32 with a 32-bit Menu file allows both 32-bit and 16-bit frames to be loaded from a 32-bit Menu file. When $MENU32 is running with a 32-bit Menu file the above table becomes:

32-bit Menu Handler ($MENU32) with 32-bit Menu file				
Menu line type	**Run in 16-bit flag**	**Program/frame type**	**Result**	**Notes**
S	Not available	16-bit program	OK	1
S	Not available	32-bit program	OK	1
F	Y	16-bit frame	OK	2
F	Y	32-bit frame	ILL OPC AT 2070	3
F	N	16-bit frame	STOP 25015	2
F	N	32-bit frame	OK	3
E	Merged with type "F" in 32-bit Menu files			
Others	Not available	Not applicable	See $MH doc'n	

Note-1 Both 16-bit programs and 32-bit programs (i.e. applications that do not require open Speedbase databases) can invoked from type "S" menu lines by $MENU32 without any special precautions.

Note-2 **The combination of the 32-bit Menu Handler, $MENU32, and a 32-bit Menu file can be used to invoke a 16-bit Speedbase frame providing the "Run frame in 16-bit mode" flag is set to "Y"**.

Note-3 The setting of the "Run frame in 16-bit mode" flag is NOT ignored by $MENU32. This flag must be set to "N" for 32-bit frames.