RAND McNALLY

MileMaker® OS/MVS

MILEMAKER®

Technical Guide

MileMaker® OS/MVS

MILEMAKER®

Technical Guide

© Rand McNally & Company
8255 North Central Park Avenue
Skokie, IL 60076-2970
Phone: (800) 234-4069
www.trucking.randmcnally.com

Table of Contents

GENERAL INFORMATION

Chapter 1

Chapter Contents

Introduction

MileMaker, a Rand McNally & Company system, quickly and accurately provides Household Goods Mileage Guide (HHG) rating mileages. Correct and consistent mileages, determined in accordance with the Household Goods Mileage Guide rules, are obtained simply by entering an origin and a destination.

The MileMaker system is available in both online, a service which affords the user instant responses at a terminal, and batch, a means of submitting requests for distances in volume. Both methods also have an optional subroutine available to the user which allows the user to control the output by interfacing his programs with the ones provided by Rand McNally & Company.

For the client installing this package, it is helpful to have someone who has a technical background, knowledge of IBM JCL, specific knowledge in the use of IDCAMS for creating files, and an understanding on how to bring up a CICS system.

Note: Review all JCL and change it according to your own shop standards, especially the attributes used when defining the VSAM clusters. Your standards may differ from Rand McNally's. In all cases Rand McNally attempts to use general parameters that apply at most customer sites. The minimum operating environment requirements are as follows:

- IBM z/0S V1.7 operating system
- IBM CICS Transaction Server V2.3
- IBM Enterprise COBOL
- Approximately 380 megabytes of disk storage (450 cylinders on a 3390 device). Additional space may be required for optional user updatable files. Approximately an additional 225 megabytes of disk storage (313 cylinders) is required if previous guide is supported.

Initial Installation Guide

Chapters 1–6 describe the technical aspects of installing the MileMaker system. Each of these chapters deal with a specific installation topic. Also included in Appendix A are lists and worksheets that contain pertinent information about the installation of the system.

Chapter 1 contains general information concerning the use of this guide and the contents of the installation tapes.

Chapter 2 contains a step by step checklist to be used in the installation of the MileMaker system.

Chapter 3 provides detailed instructions for the installation of files which may require special handling. These are user updateable files. Special procedures are required in case the user wishes to preserve data which he added to the file.

Chapter 4 contains descriptions of the optional MileMaker files. The descriptions are intended to help the client decide whether or not the files will be used.

Chapter 5 concerns the Parameter File (PM010D). Information regarding establishment of user passwords is included.

Chapter 6 provides descriptions of the various batch programs available in MileMaker.

Appendix A includes the **Current Guide Dataset Name Worksheet** and the **Previous Guide Dataset Name Worksheet**, which is provided to help keep a record of data set names that you will change from our naming conventions to yours. We suggest that you complete this form prior to modifying any JCL. Also, you may modify any of the IDCAMS options to suit your own standards.

Installation Tapes

Depending on which package has been purchased, two or more tapes will be distributed for the initial installation of the MileMaker system. The cover letter that accompanied this document will identify which tape(s) contains JCL, tables, load and source modules (SYSTEM DISTRIBUTION TAPE) and which tape(s) contains the MileMaker data file (DATA DISTRIBUTION TAPE).

The "SYSTEM DISTRIBUTION TAPE" contains PDS files that were created using the IBM utility IEBCOPY. Therefore, the same utility must be used to copy them from our tape to your DASD device. This tape contains the following:

Label	DSN	System Distribution Tape Contents
01	TDM.JCL.T	This PDS contains all JCL required in the installation operation of the system. See Appendix A for list of members.
02	TDM.TBL.T	This PDS contains all the CICS Table entries required for the CICS system. See Appendix A for list of members
03	TDM.LDL.T	This PDS contains all the load modules for the CICS and/or BATCH system(s). See Appendix A for list of members
04	TDM.SRC.T	This PDS contains all the source modules for the CICS and/or BATCH system(s)

Minimum Space And DCB Info For PDS

DSN	I/O	DSORG	RECFM	LRECL	BLKSIZE	ALLOC	DIRECTORY
TDM.JCL.T	I	PO	VS	18108	18112	6 TRKS	20
	O		FB	80	9040		
TDM.TBL.T	I	PO	VS	18108	18112	2 TRKS	20
	O		FB	80	9040		
TDM.LDL.T	I	PO	VS	19085	19089	32 TRKS	50
	O		U	*	19069		
TDM.SRC.T	I	PO	VS	18108	18112	450 TRKS	50
	O		FB	80	9040		

*no record size is specified if RECFM is U

When loading down Rand McNally's system tapes it is important to set up your JCL correctly to ensure that you load down all the data on the tape. You must use the IBM utility IEBCOPY to properly load the data. The example below shows how to set up your JCL. The space requirements in this example are for a 3380 disk.

```
//TDMDATA JOB
//************************************************************
//** STEP01 COPY PDS TO YOUR SYSTEM
/************************************************************
//STEP01   EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//JCLIN    DD DSN=TDM.JCL.T,       -TDM JCL MEMBERS
//         DISP=(OLD,KEEP,KEEP),
//         UNIT=TAPE,
//         LABEL=1,
//         VOL=SER=000000 -CHANGE TO PROPER VOLSER
//TBLIN    DD DSN=TDM.TBL.T,       -TDM TABLE MEMBERS
//         DISP=(OLD,KEEP,KEEP),
//         UNIT=TAPE,
//         LABEL=2,
//         VOL=SER=000000          -CHANGE TO PROPER VOLSER
//LDLIN    DD DSN=TDM.LDL.T,       -TDM LOAD MEMBERS
//         DISP=(OLD,KEEP,KEEP),
//         UNIT=TAPE,
//         LABEL=3,
//         VOL=SER=000000          -CHANGE TO PROPER VOLSER
//SRCIN    DD DSN=TDM.SRC.T,       -TDM SOURCE MEMBERS
//         DISP=(OLD,KEEP,KEEP),   -(OPTIONAL)
//         UNIT=TAPE,
//         LABEL=4,
//         VOL=SER=000000          -CHANGE TO PROPER VOLSER
//JCLOUT   DD DSN=(YOUR JCL LIBRARY NAME),
//         DISP=(NEW,CATLG,DELETE),
//         UNIT=SYSDA,
//         SPACE=(TRK,(10,,20),RLSE),
//         DCB=(RECFM=FB,LRECL=80,BLKSIZE=9040,DSORG=PO)
//TBLOUT   DD DSN=(YOUR TABLE LIBRARY NAME),
//         DISP=(NEW,CATLG,DELETE),
//         UNIT=SYSDA,
//         SPACE=(TRK,(10,,20),RLSE),
//         DCB=(RECFM=FB,LRECL=80,BLKSIZE=9040,DSORG=PO)
//LDLOUT   DD DSN=(YOUR LOAD LIBRARY NAME),
//         DISP=(NEW,CATLG,DELETE),
//         UNIT=SYSDA,
//         SPACE=(TRK,(90,,25),RLSE),
//         DCB=(RECFM=U,LRECL=0,BLKSIZE=19069,DSORG=PO)
```

```
//SRCOUT   DD DSN=(YOUR SOURCE LIBRARY NAME),
//         DISP=(NEW,CATLG,DELETE),
//         UNIT=SYSDA,
//         SPACE=(TRK,(450,,50),RLSE),
//         DCB=(RECFM=FB,LRECL=80,BLKSIZE=9040,DSORG=PO)
//SYSUT3   DD UNIT=SYSDA,SPACE=(TRK,(1))
//SYSUT4   DD UNIT=SYSDA,SPACE=(TRK,(1))
//SYSIN    DD *
  COPY OUTDD=JCLOUT,INDD=JCLIN
  COPY OUTDD=TBLOUT,INDD=TBLIN
  COPY OUTDD=LDLOUT,INDD=LDLIN
  COPY OUTDD=SRCOUT,INDD=SRCIN
/*
//
```

The Data DistributionTapes contain data files for the MileMaker system. These tapes contain the files listed below. The files are sequential in format.

Data Distribution Tape File Attributes

DSN	RECFM	LRECL	BLKSIZE	CONTENTS
TDM.PM010D.T	FB	80	32000	Parameter File
TDM.GE010D.T	FB	124	32240	Geographic File
TDM.MI140B.T	FB	4088	28616	Mileage/Route File
TDM.MI145B.T	FB	8	32000	Traversal File
TDM.GE020D.T	FB	146	32704	Geographic Update File
TDM.MI015D.T	FB	2000	30000	ZIP Code Mileage File
TDM.MI020D.T	FB	2036	30540	Locator File
TDM.MI105D.T	FB	31	31000	ZIP Code Master File
TDM.MI305D.T	VB	2211	31204	Blocker File
TDM.MI340M.T	VB	22	31504	Short Distance Mileage (AKB) File
TDM.MI350D.T	FB	9	27000	Cross Reference File
TDM.MI605M.T	VB	947	32000	Secondary Keypoint Mileage File
TDM.UM010D.T	VB	265	26504	User Conversion File
TDM.MI390D.T	VB	46	50	Component Route File
TDM.MI520D.T	FB	60	60	Billable Point Pairs File
TDM.MI882D.T	FB	60	32700	5-Digit ZIP Reference File

Note: For all files with RECFM=VB, LRECL=MAXIMUM LRECL in IDCAMS define + 4 byte LLBB necessary for tape.

Contents of MileMaker MVS Guide 19 Distribution on IBM 3490 Tape

The contents of the Guide 19 MileMaker MVS software and data tapes containing Basic HHG Mileage, Practical Routing, and Shortest Routing are as follows:

TAPE VOLSER	FILENO	DATASET	RECORD COUNTS	TAPE LABEL
CT500 (Source)	1	TDM.JCL.T	JCL Members	**1 of 2 GDE19MM**
	2	TDM.TBL.T	CICS Tables	
	3	TDM.LDL.T	Enterprise Cobol Load Members	
	4	TDM.SRC.T	Source Members	
CT501 (Basic HHG Files)	1	TDM.MI140B.T	36,792	**2 of 2 GDE19MM**
	2	TDM.MI145B.T	9,868	
	3	TDM.MI340M.T	4,823,298	
	4	TDM.MI305D.T	51,610	
	5	TDM.GE010D.T	154,282	
	6	TDM.MI020D.T	383	
	7	TDM.GE020D.T	1	
	8	TDM.UM010D.T	1	
	9	TDM.MI350D.T	6,132	
	10	TDM.MI605M.T	1	
	11	TDM.PM010D.T	104	
	12	TDM.MI390D.T	1	
	13	TDM.MI520D.T	1	
	14	TDM.MI015D.T	916	
	15	TDM.MI105D.T	993	
	16	TDM.MI882D.T	150,332	

TAPE VOLSER	FILENO	DATASET	RECORD COUNTS	TAPE LABEL
CT502 (Practical Files)	1	TDM.MI321D.T	618,531	**1 of 1 GDE19PR**
	2	TDM.MI141B.T	18,396	
	3	TDM.MI306D.T	79,921	
	4	TDM.MI021D.T	567	
	5	TDM.MI351D.T	6,132	
	6	TDM.MI605R.T	1	
	7	TDM.MI391D.T	1	
	8	TDM.MI341R.T	4,552,979	
CT503(Shortest Route Files)	1	TDM.MI340R.T	4,823,298	**1 of 2 GDE19SH**
CT504	1	TDM.MI320D.T	1,376,915	**2 of 2 GDE19SH**
	2	TDM.MI605R.T	1	

Within thirty days of completion of the installation, please return the distribution tapes to:

Rand McNally & Company
8255 North Central Pk. Ave.
Skokie, IL 60076
ATTN: Rand McNally Distribution

If you have any questions, please call Rand McNally & Company's Customer Service at (800) 234-4069.

Technical Considerations

SHAREOPTION

The FCT entry for updateable MileMaker files is distributed as (2 3). This is to prevent the files from being damaged by simultaneous updating from two different regions. The following should be considered as a consequence of this SHAREOPTION setting:

- If the optional Billing File (MI520D) is used, two separate Billing files are required, one for batch and one for on-line.
- Currently the optional Component file (MI390D) is not updated by the batch system if the on-line system is active. It is opened for inquiry only if the on-line system is active. Therefore, the on-line Component file MI390D must be closed if the batch Component file delete program is run.

SHAREOPTION for non-updateable MileMaker files is distributed as (3 3) since no cross-region updating is possible on these read-only files.

Compile Options

If recompilation is required of either MileMaker or user-written interface modules, then one of the following compile options listed below must be used. All MileMaker distributed load modules were compiled using Enterprise COBOL for Z/05 using the following compile options:

```
CICS COBOL: RENT, NODYNAM, TRUNC(BIN), NUMPROC(MIG), NOCMPR2,
NOOPT, DATA(31)

BATCH COBOL: RENT, DYNAM, TRUNC(BIN), NUMPROC(MIG), NOADV,
NOCMPR2, NOOPT, DATA(31)
```

If a MileMaker BMS map assembly is required, it must be assembled as an unaligned map using the option SYSPARM(MAP) for load modules.

File Placement

If MileMaker is being installed on more than one DASD volume, the following performance considerations can be used.

- The base cluster of the Geographic file (GE010DP) should be allocated on a different volume than the alternate index datasets (GE010D1 and GE010D2). Also the index components of each of these datasets can be separated from the data components to reduce seek time. See Appendix A for dataset and component names used in the distribution JCL.
- If possible, the data components of the Geographic file (GE010DP) and the Mileage Route file (MI140B) should be located on separate volumes.

MileMaker High Volume Recommendations

If a high volume of MileMaker transactions is expected, then the following performance tuning procedures are recommended:

1. Place all MileMaker files in VSAM LSR. Less file buffers will be required and file I/O will be reduced due to buffer look-aside.
2. The two most active MileMaker files, MI140B, GE010DP, should be placed on separate disk volumes.

3. Increase the STRNO in the MileMaker FCT entries of the most active files to the expected number of concurrent transactions.
4. Increase the number of index buffers for the most active files.
5. An increase in the CICS region size may be required in order to give CICS more storage for the additional transactions.
6. Place the MileMaker transactions into a special CICS transaction class with a limit on the number of concurrent transactions. This will insure that MileMaker will not monopolize the resources in a CICS region, especially storage and file I/O. This is implemented by using the TCLASS parameter on the PCT.

MILEMAKER INSTALLATION

Chapter 2

Chapter Contents

Introduction

This section contains a step-by-step checklist which can be used as a guide for the installation of the Guide 19 Release of BASIC MileMaker.

All procedures, CICS tables and load modules referred to in this checklist can be found on the distribution tape. The member name is the same as the procedure name. See Chapter 1 of this document for further information regarding the distribution tape.

The files distributed with Release 19 replace your current files, if any. Guide 18 files are NOT being provided with Release 19. If previous Guide support is desired, your Guide 18 files will be renamed to replace your previous Guide files.

Each file create job contains IDCAMS commands to delete the dataset, create the dataset, and load the dataset. The Geographic File(GE010D) has two alternate indexes. The create procedure for this dataset contains IDCAMS commands to define and build the alternate indexes and paths. When checking completion codes, note that the delete operation will return a completion code of 08 if the dataset does not exist. All other operations should return a completion code of zero (00).

The modules referred to in Steps 17 and 21 of the installation checklist are contained in the SYSTEM DISTRIBUTION TAPE on dataset 'TDM.LDL.T'. A list of the programs required for Basic MileMaker can be found in Appendix A. After the SYSTEM DISTRIBUTION TAPE is loaded onto your system (see Chapter 1), these load modules should be moved to the CICS or BATCH APPLICATION LOAD LIBRARY.

The source code is also supplied in case recompilation is required. The source code can be found on the SYSTEM DISTRIBUTION TAPE on dataset 'TDM.SRC.T'. These members are not needed unless recompilation is required. If recompilation is required, refer to the **Compile Options** section of Chapter 1.

Chapter 3 contains instructions for installing files which may require special handling. These are all user-updateable files. Special handling is required to preserve any user data on these files.

See Chapter 4 for a description of the optional files referred to in Steps 11 through 15.

See Chapters 3 and 5 for further information regarding Parameter File installation.

Installation of MileMaker Guide 19 OS/MVS Library/Data Files from CD-ROM

Please read these supplemental instructions for loading the MileMaker application installation from CD-ROM. Once the initial load is complete, you may consult the OS/MVS Technical Guide in PDF format that is included on your Rand McNally Documentation CD.

All files on the CD-ROM must be uploaded from a PC to the mainframe using FTP or a PC3270 file transfer utility using IND$FILE. All file uploads must be performed in binary mode.

Creation of the MileMaker MVS Libraries/Partitioned Files

The MileMaker libraries/PDSs on the CD-ROM are in XMIT format that were created with the TSO TRANSMIT command.

There is a JCL library called MM19.MVS.JCLLIB.XMIT, a source library called MM19.MVS.SRCLIB.XMIT, a load library called MM19.MVS.LOADLIB.XMIT, and a CICS table library called MM19.MVS.TBLLIB.XMIT.

1. Copy the XMIT library files from CD-ROM to your PC's hard drive.
2. Before uploading, set the mainframe file attributes of each file to the following: LRECL=80, RECFM=FB, BLKSIZE=3120, CYL=xxxxx, where xxxxx is a suitable amount of disk space in cylinders.
3. Perform a binary upload of the library XMIT library files on the PC to the mainframe.
4. On the mainframe perform a TSO RECEIVE on each of the XMIT library files to produce the MileMaker libraries. Refer below to the sample JCL that illustrates the TSO RECEIVE process.

Creation of the MileMaker MVS Data Files

The MileMaker fixed and variable length files on the CD-ROM are in XMIT format and identified as 'MM19.MVS.xxxxxx.XMIT' where xxxxxx is the base file name. There are 26 MileMaker MVS data files. The following two files have been compressed: MM19.MVS.MI340R.XMIT.ZIP and MM19.MVS.MI341R.XMIT.ZIP.

1. Copy the MileMaker data XMIT files to your PC's hard drive.
2. Before uploading set the mainframe file attributes of each file to the following: LRECL=80, RECFM=FB, CYL=xxxxx, where xxxxx is a suitable amount of disk space in cylinders.

3. Perform a binary upload of the XMIT data files on the PC to the mainframe.
4. On the mainframe, perform a TSO RECEIVE on each of the XMIT data files to produce the MileMaker fixed and variable length data disk files. Refer below to the sample TSO RECEIVE JCL.
5. Follow the installation steps in the Technical Guide to create the MileMaker VSAM files from the disk data files created by TSO RECEIVE.

Sample Batch TSO RECEIVE JCL

```
//RECV  JOB
//RECV  EXEC  PGM=IKJEFT01
//SYSTSPRT  DD  SYSOUT=A
//SYSTSIN   DD  *
RECEIVE INDSN('userid.jcllib.xmit')
DSN('userid.jcllib.pds')
/*
//
```

Guide 19 Installation Checklist

Use the following checklist to install Guide 19. It is assumed that all Guide 19 files and modules are initially placed into a test environment for verification.

Note that if you are using only Milemaker Practical and HHG 3-Digit ZIP, then perform ONLY the following installation Steps: 1, 2, 3, 10, 13, 14, 15, 17, 18, 19, 20, 21, 22.

- ☐ **STEP 1** Unload SYSTEM DISTRIBUTION TAPE or CD-ROM.
- ☐ Unload "TDM.JCL.T" from SYSTEM DISTRIBUTION TAPE.
- ☐ Unload "TDM.TBL.T" from SYSTEM DISTRIBUTION TAPE.
- ☐ Unload "TDM.LDL.T" from SYSTEM DISTRIBUTION TAPE.
- ☐ Unload 'TDM.SRC.T" from SYSTEM DISTRIBUTION TAPE.

- ☐ **STEP 2** Build and load USER CONVERSION FILE.
- ☐ See special instructions in Chapter 3.

- ☐ **STEP 3** Build and load GEOGRAPHIC FILE.
- ☐ See Chapter 3 for instructions.

- ☐ **STEP 4** Build and load MILEAGE/ROUTE FILE.
- ☐ Modify MI140BVD procedure to your standards.
- ☐ Execute MI140BVD procedure.
- ☐ Check return codes.

- ☐ **STEP 5** Build and load TRAVERSAL FILE.
- ☐ Modify MI145BVD procedures to your standards.
- ☐ Execute MI145BVD procedure.
- ☐ Check return codes.

- ☐ **STEP 6** Build and load LOCATOR FILE.
- ☐ Modify MI020DVD procedure to your standards.
- ☐ Execute MI020DVD procedure.
- ☐ Check return codes.

☐ **STEP 7** Build and load BLOCKER FILE.

☐ Modify MI305DVD procedure to your standards.

☐ Execute MI305DVD procedure.

☐ Check return codes.

☐ **STEP 8** Build and load SHORT DISTANCE MILEAGE (A-K-B) FILE.

☐ Modify MI340MVD procedure to your standards.

☐ Execute MI340MVD procedure.

☐ Check return codes.

☐ **STEP 9** Build and load CROSS-REFERENCE FILE.

☐ Modify MI350DVD procedure to your standards.

☐ Execute MI350DVD procedure.

☐ Check return codes.

☐ **STEP 10** Build and load PARAMETER FILE.
See Chapter 3 for instructions.

☐ **STEP 11** Build and load ROUTE COMPONENT FILE. (OPTIONAL)
See Chapter 4 for file description.

☐ Modify MI390DVD procedure to your standards.

☐ Execute MI390DVD procedure.

☐ Check return codes.

☐ **STEP 12A** Existing transaction-based customers must build and load the BILLING FILE using the G18 (Guide 18) BILLING FILE.

☐ Modify MI520DVR procedure to your standards.

☐ Execute MI520DVR procedure.

☐ Check return codes.

☐ Proceed to STEP 13.

☐ **STEP 12B** New transaction-based customers must build and load a new BILLING FILE.

☐ Modify MI520DVD procedure to your standards.

☐ Execute MI520DVD procedure.

☐ Run the MIBILLX procedure. See User Manual for details.

☐ Check return codes.

☐ Proceed to STEP 13.

☐ **STEP 12C** All other customers may optionally build and load the BILLING FILE.

☐ Modify MI520DVD procedure to your standards.

☐ Execute MI520DVD procedure.

☐ Check return codes.

☐ **STEP 13** Build and load GEOGRAPHIC UPDATE FILE. (OPTIONAL) See Chapter 4 for file description.

☐ Modify GE020DVD procedure to your standards.

☐ Execute GE020DVD procedure.

☐ Check return codes.

☐ **STEP 14** Build and load ZIP CODE MILEAGE FILE. (OPTIONAL)*
See Chapter 4 for file description.

☐ Modify MI015DVD procedure to your standards.

☐ Execute MI015DVD procedure.

☐ Check return codes.

☐ **STEP 15** Build and load ZIP CODE MASTER FILE. (OPTIONAL)*
See Chapter 4 for file description.

☐ Modify MI105DVD procedure to your standards.

☐ Execute MI05DVD procedure.

☐ Check return codes.

* If MI015D and MI105D files were loaded for Release 18, it is not necessary to reload them.

☐ **STEP 16** Build and load the 5-digit ZIP REFERENCE FILE.

☐ Modify MI882DVD procedure to your standards.

☐ Execute MI882DVD procedure.

☐ Check return codes.

☐ **STEP 17** CICS Tables.

☐ Modify MIDDS to your standards.

☐ Add DD statements from MIDDS to CICS start-up JCL.

☐ Add MIFCT statements to your FCT table.

☐ Add MIPCT statements to your PCT table.

☐ Add MIPPT statements to your PPT table.

☐ Assemble FCT table.

☐ Assemble PCT table.

☐ Assemble PPT table.

☐ **STEP 18** CICS Modules.

☐ Load CICS modules to your CICS application library.

☐ **STEP 19** Activate or deactivate optional files using the parameter file update option. Refer to the User Manual.

☐ **STEP 20** Activate or deactivate MileMaker internal security using the parameter file update option. Refer to the User Manual.

☐ **STEP 21** Build and load SECONDARY KEYPOINT FILE. See Chapter 3 for instructions.

☐ **STEP 22** BATCH Modules.

☐ Load BATCH modules to your BATCH application library.

☐ **STEP 23** OPTIONAL BATCH Processing. (Refer to Chapter 6 for details.)

☐ Load OPTIONAL BATCH modules to your BATCH application library.

☐ Modify MI200X procedure to your standards.

☐ Execute MI200X procedure.

☐ Check return codes and reports.

☐ Modify MI921X procedure to your standards.

☐ Execute MI921X procedure.

☐ Check return codes.

☐ Modify MI951X procedure to your standards.

☐ Execute MI951X procedure.

☐ Check return codes.

☐ Modify MI210X DD statements to your standards.

☐ Add MI210X DD statements to procedure for user calling program.

☐ Execute user procedure.

☐ Check return codes and reports.

Previous Guide 18 Installation

In general, Production Guide 18 files will be copied to newly defined Guide 19 previous Guide files. The following steps are required only if the client wishes to support Guide 18. Previous Guide files can be deactivated using the parameter file update option from the CICS MileMaker main menu. New customers will not receive previous Guide 18 files and will need to deactivate previous Guide.

- ☐ **STEP 1** Copy the Production G18 MI140B file to the newly defined G19 MI140A file.
- ☐ Modify MI140AVD procedure to your standards
- ☐ Execute MI140AVD procedure.
- ☐ Check return codes.
- ☐ **STEP 2** Copy the Production G18 MI145B file to the newly defined G19 MI145A file.
- ☐ Modify MI145AVD procedure to your standards.
- ☐ Execute MI145AVD procedure.
- ☐ Check return codes.
- ☐ **STEP 3** Copy the Production G18 MI020D file to the newly defined G19 MI022D file.
- ☐ Modify MI022DVD procedures to your standards.
- ☐ Execute MI022DVD procedure.
- ☐ Check return codes.
- ☐ **STEP 4** Copy the Production G18 MI305D file to the newly defined G19 MI307D file.
- ☐ Modify MI307DVD procedure to your standards.
- ☐ Execute MI307DVD procedure.
- ☐ Check return codes.
- ☐ STEP 5 Copy the Production G18 MI340M file to the newly defined G19 MI342M file.
- ☐ Modify MI342MVD procedure to your standards.
- ☐ Execute MI342MVD procedure.
- ☐ Check return codes.

☐ **STEP 6** Copy the Production G18 MI350D file to the newly defined G19 MI352D file.

☐ Modify MI352DVD procedure to your standards.

☐ Execute MI352DVD procedure.

☐ Check return codes.

☐ **STEP 7** Copy the Production G18 GE010DP file to the newly defined G19 GE011DP previous Guide geographic file. GE011DP, GE011D1, GE011D2 are new files for Guide 19.

☐ Modify GE011DVD procedure to your standards.

☐ Execute GE011DVD procedure.

☐ Check return codes.

☐ Add GE011DP file to your CICS system. Use GE010DP file as a model.

☐ Add GE011D1 file to your CICS system. Use GE010D1 file as a model.

☐ Add GE011D2 file to your CICS system. Use GE010D2 file as a model.

☐ **STEP 8** Copy the production G18 MI882D file to the newly defined G19 previous Guide MI883D file. The MI883D file is a new file for Guide 19

☐ Modify MI883DVD procedure to your standards.

☐ Execute MI883DVD procedure.

☐ Check return codes.

☐ Add MI883D file to your CICS system. Use MI882D file as a model.

☐ **STEP 9** Build and load the previous Guide Secondary Keypoint Mileage file. See Chapter 3 concerning instructions for loading previous Guide Secondary files.

SPECIAL FILE CREATION PROCEDURES

Chapter 3

Chapter Contents

UM010D–User Conversion File

Two options are available for loading the User Conversion File. New clients should use option 1.

Option 1: If the user has not added records to his UM010D file the following steps should be taken:

- ☐ Modify UM010DVD procedure to your standards.
- ☐ Execute UM010DVD procedure.
- ☐ Check return codes. (A single dummy record will be loaded)

The UM010DVD procedure deletes, redefines and loads the User Conversion File.

Option 2: If the user has added records to the UM010D File follow these steps:

- ☐ No special processing is necessary. No User Conversion records were distributed for Guide 19.

Geographic File (GE010D)

Two options are available to the user in order to establish the Release 19 Geographic File (GE010D). New clients should use Option 1.

Option 1: If the user has not added records to the Guide 18 GE010D, the Rand McNally & Co distributed file is loaded as distributed. Follow these steps:

- ☐ Modify GE010DVD procedure to your standards
- ☐ Execute GE010DVD procedure
- ☐ Check return codes.

Upon completion of this step, you may now proceed to the next step in the checklist in the Guide 19 Installation Checklist section of Chapter 2.

Option 2: If the user has added records to the GE010D and wishes to preserve those records, the following procedure must be followed. This procedure merges the Guide 18 GE010D with the Guide 19 GE010D from Rand McNally & Co. Follow these steps:

- ☐ Modify GE965X procedure to your standards and as described below.
- ☐ Execute GE965X procedure.
- ☐ Check return codes and reports.
- ☐ Modify GE010DVD procedure as described in this section.

☐ Execute GE010DVD procedure

☐ Check return codes.

Note: the remainder of this section provides a more detailed description of this process.

Upon completion of this step, you may now proceed to the next step in the checklist in the Guide 19 Installation Checklist section of Chapter 2.

GE965 Procedure

The GE965X procedure executes the GE965 program. This program is provided on dataset "TDM.SRC.T" on the System Distribution Tape. A description of the procedure follows:

Input:

DDNAME	Type	Description
GE010DP	VSAM	Guide 18 Geographic File
GE010T	SEQ	Guide 19 Geographic File from Rand McNally Data Distribution Tape

Output:

DDNAME	Type	Description
GE010N	SEQ	Merged Geographic File for use in Guide 19. This file is used as input to the GE010DVD procedure.
UM010D	VSAM	Guide 19 User Conversion File

This procedure has two steps. The first step copies the VSAM GE010D which is being used for Guide 18 processing to a sequential backup file. Step 2 merges the Rand McNally distributed Guide 19 GE010D with the copy of the user's GE010D created in step one. Any record on the Guide 18 GE010D which has the same key (city name, SPLC, or internal key) as a record on the Guide 19 GE010D will be dropped from the output file.

Reports

Six reports are printed by the GE965 program. Descriptions of these reports follow.

Data Field Changes: This report lists records which have the same name on the customer file and the Rand McNally file and have differences in certain fields. An asterisk to the right of field indicates that the value is different on the customer and Rand McNally files. For the SPLC, keypoint code and keypoint number, the Rand McNally value is always used on the combined GE010D file. The Rand McNally

longitude and latitude are used only if they are valid. Some records have a longitude of zero or 100.00 and a latitude of zero or 50.00.

A record is written to the User Conversion File (UM010D) for every SPLC change so that the customer may continue to use the old SPLC.

City Name Changes: This report lists records which have the same SPLC on the Rand McNally and customer files but have different names. The Rand McNally name is always used on the combined GE010D file. A record is written to the User Conversion File (UM010D) for each city name change so that the customer may still use the old name. Other field value differences are treated as in the DATA FIELD CHANGES report described above.

User Conversion SPLC Records: This report lists the user conversion SPLC records written to the UM010D file. There is a record written for each SPLC change listed in the DATA FIELD CHANGES report. The Rand McNally state, city and SPLC are listed along with the customer's old SPLC, which is the user code on the user conversion record. If a write fails there will be a message to the right of the user code.

User Conversion City Records: This report lists user conversion city records written to the UM010D file. There is a record written for each line on the CITY NAME CHANGES report. The report lists the Rand McNally state, name and SPLC and the customer's city name, which is the user code on the UM010D record. If a write fails, a message will appear to the right of the user code.

User Records with Bad Long/Lats: This report lists all customer records which do not have a matching name on the Rand McNally Geographic File and have an invalid longitude or latitude. An invalid longitude is zero or 100.00. An invalid latitude is 50.00 or zero. These records are not written to the combined Geographic File.

Dup Internal Key Dropped: Rand McNally uses an 'internal key' as a key to accessing most of the MileMaker files. This key normally is the same as the SPLC but may not be because the SPLC can change but the internal key cannot. Since the internal key is used as a Geographic File alternate key there can be no duplicate internal keys. Therefore, a customer record is dropped if it has the same internal key as a record on the Rand McNally Geographic File. These records are listed on this report.

Customer Records with 99 SPLC's Deleted: This report lists customer records which are erroneous or no longer used. These records have an SPLC beginning with "99" and an "internal key" beginning with "99". These records are not written to the combined Geographic File. The report lists the records' city name, SPLC, longitude and latitude.

The following control totals are printed at the conclusion of the GE965 program. They appear immediately after the **Dup Internal Key Dropped** report.

Customer geographic record read	Rand McNally distribution records read
Unmatched customer geo records	Unmatched Rand McNally distribution records
Matched geographic records	City name changes
Geographic records written	Matching internal keys dropped
99 Guide 18 SPLC dropped	Customer bad long/lats dropped
UM010D city records added	UM010D city records changed
UM010D SPLC records added	UM010D SPLC records changed

The above control totals can be balanced as follows:

1. Customer geographic records read =

   Unmatched customer geo records

   + Matched geographic records

   + Cust Guide 18 99 SPLC dropped

   + Customer bad long/lat dropped

2. Rand McNally & Co distribution records read =

   Unmatched Rand McNally distribution records

   + Matched geographic records

3. Geographic records written =

   Unmatched Rand McNally & Co distribution records

   + Unmatched customer geo records

   + Matched geographic records

   – Matching internal keys dropped

   – City name changes

GE010DVD Procedure

Upon completion of the GE965X job, the merged GE010D is ready to load. The GE010T DDNAME statement in the GE010DVD procedure should be changed to point to the merged Geographic File created by the GE965X procedure (DDNAME name GE010NEW).

The user may make other modification to the procedure as needed. After the required modification, execute the procedure, check the results and proceed to the next step in the checklist.

PM010D - Parameter File

Two options are available for loading the MileMaker Parameter File, PM010D. New clients must use Option 1. Existing clients should use Option 2.

Option 1: The Rand McNally distributed parameter file will be loaded directly from the DATA DISTRIBUTION TAPE. Please follow these steps:

- ☐ Modify the PM010DVD procedure to your standards.
- ☐ Execute the PM010DVD procedure.
- ☐ Check return codes (see below).

If the return codes are zero, continue with the next step in the installation checklist in Chapter 2.

The parameter file can be used to add user passwords and user ids to the MileMaker system.

Option 2: The following procedure will copy existing Guide 18 PM010D to a newly defined Guide 19 PM010D leaving any user added security records intact.

- ☐ Modify the PM010DVR procedure to your standards.
- ☐ Execute the PM010DVR procedure.
- ☐ Check return codes (see below).

If the return codes are zero, continue with the next step in the installation checklist in Chapter 2.

MI605M/R - Secondary Keypoint Files

The current guide Secondary Files are the Secondary Keypoint Mileage File, MI605M, and the Secondary Keypoint Route File, MI605R. Only the MI605M file is required for Basic MileMaker. The MI605R file is required for Shortest and Practical Routing. The same MI605R file will be used for both Shortest and Practical Routing. No new Secondary Mileage records are distributed for Guide 19.

Two optional procedures are provided in Basic MileMaker in order to load the current guide Secondary Keypoint Mileage File, MI605M. New clients must use procedure 1. Existing clients should use procedure 1 only if users have not added

their own secondary records. If users have added secondary records, or if the Geographic merge step (GE965X) was run , procedure 2 must be used.

Option 1: The Rand McNally & Co distributed current guide Secondary Keypoint Mileage File will be loaded directly from the DATA DISTRIBUTION TAPE. Refer to the Shortest or Practical Routing install guides for instructions on how to install the MI605R file. Please follow these steps:

- ☐ Modify MI605MVD procedure to conform to your standards
- ☐ Execute MI605MVD procedure.
- ☐ Check return codes
- ☐ Modify MI605MVR procedure to conform to your standards
- ☐ Execute MI605MVR procedure
- ☐ Check return codes (A single MI605M dummy record will be loaded)

Unlike other file loads, this is done in two steps. The MI605MVD procedure defines the file. The MI605MVR procedure loads the records. The user may combine these procedures if he desires.

If return codes are valid, continue with the next step in the checklist in Chapter 2.

Option 2: Existing clients must run an edit/update procedure, MI9605X, against their current guide secondary files to insure that the geographic related fields are in sync with the Geographic File, GE010D. The secondary files will be updated if they are out of sync with GE010D. Another procedure, MI605MVR, must be run which will merge Rand McNally secondary records into the secondary files.

Since the MI9605X procedure is capable of creating records for the Secondary Keypoint Route File (MI605R) the following steps include creation of the MI605R file. Skip these steps only if routing is not used. Also, if routing is not used, make sure that the Shortest Route and Practical Route flags on the Parameter File (PM010D) MI115 record are set to 'N'. The PM010D file distributed by Rand McNally contains a 'Y' in these fields. See the USER MANUAL for further information concerning these flags.

In order to edit/update your current guide secondary files please follow the steps below. The steps marked with an asterisk (*) should be skipped if the client is not using routing. Note: the MI605R file is used for both Shortest Distance and Practical Routing.

- ☐ Modify MI9605X procedure to your standards.
- ☐ Execute MI9605X procedure.
- ☐ Check return codes (See Below).

In summary, the MI9605X procedure will first backup the clients secondary files, then edit and update them to correct any geographic inconsistencies. This backup will be used to build the previous guide secondary files in the procedures MI607MVX and MI607RVX.

The MI9605X procedure is discussed in greater detail below. After running this procedure, surrounding points which the user had employed for his own secondaries may be dropped for reasons discussed below. If the user requires assistance reblocking secondaries or if there are any questions concerning this procedure, please contact your Customer Support Representative.

MI9605X Procedure

The MI9605X procedure creates a backup of the current guide secondary files, MI605M and optionally MI605R, then edits and updates them in order to insure that their geographic related fields are in sync with the Geographic File.

If either Shortest or Practical Routing is specified on the Parameter File, then the current guide Secondary Keypoint Route File, MI605R, will also be updated. (Refer to the description of the MI115 record on the PM010D file in the USER MANUAL).

Input Files

DDNAME	Type	Description
MI605M	VSAM	Current guide Secondary Keypoint Mileage File
MI605R	VSAM	Current guide Secondary Keypoint Route File (optional)
PM010D	VSAM	Parameter File
UM010D	VSAM	User Conversion File

Update Files

DDNAME	Type	Description
MI605M	VSAM	Current guide Secondary Keypoint Mileage File
MI605R	VSAM	Current guide Secondary Keypoint Route File (optional)
GE010D	VSAM	Geographic File

Output Files

DDNAME	Type	Description
OPREPORT	QSAM	Error report
MI605MK	VSAM	Secondary Keypoint Mileage File backup
MI605RK	VSAM	Secondary Keypoint Route File backup(optional)

Special Considerations

Only clients who have updated their secondary files and who wish to retain those updates should run this procedure.

This procedure must be run after loading the Geographic File (GE010D).

Secondaries which have been upgraded to nodes or keypoints will be dropped from the updated secondary files.

For Shortest and/or Practical users, routing must be specified in the Parameter File, PM010D; otherwise secondary route records will not be updated.

This procedure executes the MI9605 program. The program is contained on the SYSTEM DISTRIBUTION TAPE on dataset 'TDM.LDL.T.

Narrative

Each record on the clients secondary file will be edited and updated to insure that all geographic related fields are in sync with the Geographic File.

A secondary record will be dropped if any of the following conditions occur:

1. The secondary has no matching record on GE010D and no matching record on UM010D.
2. The secondary has a matching record on GE010D but its CURRENT keypoint code is not a '2' or ' ' (space).

A surrounding point associated with a secondary will be dropped if any of the following conditions occur:

1. The surrounding point has no matching record on GE010D and no matching record on UM010D.
2. The surrounding point has a matching record on GE010D but its CURRENT keypoint code is a '2' or ' ' (space).

CPU Requirements

This procedure will consume about one minute of CPU time.

JCL Requirements and Return Codes

The execution JCL has been provided in the member named MI9605X. This member resides on the file "TDM.JCL.T" of the SYSTEM DISTRIBUTION TAPE. Before executing, please modify the JCL to suit your installation standards.

For package clients who do not use routing, delete all references to the Secondary Keypoint Route File, MI605R and make sure that the MI115 record on the Parameter File (PM010D) specifies that routing is not available (see USER MANUAL).

Successful execution of this procedure is indicated by a return code of zero; unsuccessful execution is indicated by a return code of 016.

If there are an excessive number of VSAM control interval/area splits on either MI605M or MI605R then for performance reasons it may be advisable to reorganize them after MI9605X is run. But since the number of secondaries is small, very few VSAM control interval/area splits should occur.

Error Messages

OPEN error messages

1. MI605M OPEN ERROR - XX
2. MI605R OPEN ERROR - XX
3. GE010D OPEN ERROR - XX
4. UM010D OPEN ERROR - XX
5. PM010D OPEN ERROR - XX

Refer to OS/MVS COBOL programmers guide for an explanation of the VSAM error status codes. Also, check the JCL log for any system error messages. Two of the most common reasons for VSAM open errors are as follows:

1. A dataset was left open by a previous job that abended. If the previous job had updated the file, then a file restore is necessary.
2. The file is not available because another job has exclusive control of the file. Make sure that the VSAM Share option is (2,3) not (1,3).

Fatal File Errors

1. MI605M READ ERRORS -- XX
2. MI605M DELETE ERROR -- XX
3. MI605M REWRITE ERROR -- XX
4. MI605R DELETE ERROR -- XX

5. GE010D READ ERROR -- XX
6. GE010D REWRITE ERROR -- XX

Refer to the OS/MVS COBOL programmers guide for an explanation of the VSAM error status codes. Also, check the JCL log for any system error messages. If these errors occur, then either an I/O error occurred or a VSAM related programming error occurred. If the error code indicates a programming error, please call MileMaker Applications Development.

CLOSE error messages

1. MI605M CLOSE ERROR - XX
2. MI605R CLOSE ERROR - XX
3. GE010D CLOSE ERROR - XX
4. UM010D CLOSE ERROR - XX
5. PM010D CLOSE ERROR - XX

Refer to OS/MVS COBOL programmers guide for an explanation of the VSAM error status codes. Also, check the JCL log for any system error messages. 'XX' is the VSAM file status code.

These errors are usually the result of previous open errors.

MI607M/R - Previous Guide Secondary Files

The previous guide secondary files are the Secondary Keypoint Mileage File, MI607M, and the Secondary Keypoint Route File, MI607R. Only the MI607M file is required for Basic MileMaker. The MI607R file is required for Shortest Routing only.

New customers will skip this procedure since previous Guide 18 files are not distributed in the Guide 19 update.

Existing customers can skip this step by using the parameter update option from the CICS MileMaker main menu to deactivate previous Guide files.

Your Production Guide 18 Secondary file, MI605M, will be copied to a newly defined G19 MI607M file. Refer to the Shortest or Practical Routing Install Guides for instructions on how to install the MI607R. Please follow these steps:

- ☐ Modify MI607MVD procedure to your standards
- ☐ Execute MI607MVD procedure.
- ☐ Check return codes
- ☐ Modify MI607MVR procedure to your standards
- ☐ Execute MI607MVR procedure

☐ Check return codes

If return codes are valid, return to the checklist in Chapter 2.

MI9607X Procedure

The MI9607X procedure creates a backup of the previous guide secondary files, MI607M and optionally MI607R, then edits and updates them in order to insure that their geographic related fields are in sync with the geographic file.

If either Shortest or Practical routing is specified on the Parameter file, then the previous guide Secondary Keypoint Route File, MI607R, will also be updated. (Refer to the description of the MI115 record on the PM010D file in the USER MANUAL).

Input Files Previous Guide

DDNAME	Type	Description
MI607M	VSAM	Previous Guide Secondary Keypoint Mileage File
MI607R	VSAM	Previous guide Secondary Keypoint Route File (optional)
PM010D	VSAM	Parameter File
UM010D	VSAM	User Conversion File

Update Files Previous Guide

DDNAME	Type	Description
MI607M	VSAM	Previous Guide Secondary Keypoint Mileage File
MI607R	VSAM	Previous guide Secondary Keypoint Route File (optional)
GE010D	VSAM	Geographic File

Output Files Previous Guide

DDNAME	Type	Description
OPREPORT	QSAM	Error report
MI607MK	VSAM	Secondary Keypoint Mileage File backup
MI607RK	VSAM	Secondary Keypoint Route File backup (optional)

Special Considerations

Only clients who have updated their secondary files and who wish to retain those updates should run this procedure. This procedure must be run after loading the Geographic File (GE010D).

Secondaries which have been upgraded to nodes or keypoints will be dropped from the updated secondary files.

For SHORTEST and/or PRACTICAL users, routing must be specified in the Parameter File, PM010D; otherwise secondary route records will not be updated.

This procedure executes the MI9607 program. The program is contained on the SYSTEM DISTRIBUTION TAPE on dataset 'TDM.LDL.T.

Narrative

Each record on the clients secondary files will be edited and updated to insure that all geographic related fields are in sync with the Geographic File.

A secondary record will be dropped if any of the following conditions occur:

1. The secondary has no matching record on GE010D and no matching record on UM010D.
2. The secondary has a matching record on GE010D but its PREVIOUS keypoint code is not a '2' or ' ' (space).

A surrounding point associated with a secondary will be dropped if any of the following conditions occur:

1. The surrounding point has no matching record on GE010D and no matching record on UM010D.
2. The surrounding point has a matching record on GE010D but its PREVIOUS keypoint code is a '2' or ' ' (space).

CPU Requirements

This procedure will consume about one minute of CPU time.

JCL Requirements

The execution JCL has been provided in the member named MI9607X. This member resides on the file "TDM.JCL.T" of the SYSTEM DISTRIBUTION TAPE. Before executing, please modify the JCL to suit your installation standards.

For clients who use Basic MileMaker only, delete all references to the Secondary Keypoint Route File, MI607R and make sure that the MI115 record on the Parameter File (PM010D) specifies that routing is not available (see USER MANUAL).

Successful execution of this procedure is indicated by a return code of zero; unsuccessful execution is indicated by a return code of 016.

If there are an excessive number of VSAM control interval/area splits on either MI607M or MI607R then for performance reasons it may be advisable to reorganize them after MI9607X is run. But since the number of secondaries is small, very few VSAM control interval/area splits should occur.

Error Messages

OPEN Error Messages

1. MI607M OPEN ERROR - XX
2. MI607R OPEN ERROR - XX
3. GE010D OPEN ERROR - XX
4. UM010D OPEN ERROR - XX
5. PM010D OPEN ERROR - XX

Refer to OS/MVS COBOL programmers guide for an explanation of the VSAM error status codes. Also, check the JCL log for any system error messages. Two of the most common reasons for VSAM open errors are as follows:

1. A dataset was left open by a previous job that abended. If the previous job had updated the file, then a file restore is necessary.
2. The file is not available because another job has exclusive control of the file. Make sure that the VSAM share option is (2,3) not (1,3).

FATAL File Errors

1. MI607M READ ERROR -- XX
2. MI607M DELETE ERROR -- XX
3. MI607M REWRITE ERROR -- XX
4. MI607R DELETE ERROR -- XX
5. GE010D READ ERROR -- XX
6. GE010D REWRITE ERROR -- XX

Refer to the OS/MVS COBOL programmers guide for an explanation of the VSAM error status codes. Also, check the JCL log for any system error messages. If these errors occur, then either an I/O error occurred or a VSAM related programming error occurred. If the error code indicates a programming error, please call MileMaker Applications Development.

CLOSE Error Messages

1. MI605D CLOSE ERROR - XX
2. MI607M CLOSE ERROR - XX
3. MI607R CLOSE ERROR - XX
4. GE010D CLOSE ERROR - XX
5. UM010D CLOSE ERROR - XX
6. PM010D CLOSE ERROR - XX

Refer to OS/MVS COBOL programmers guide for an explanation of the VSAM error status codes. Also, check the JCL log for any system error messages.

These errors are usually the result of previous open errors.

OPTIONAL FILE DESCRIPTION

Chapter 4

Chapter Contents

Component File (MI390D)

The Component File contains prestored mileages and route components. When a mileage or route is requested for the first time for a pair of points, a component record is added to the file. Subsequent requests for a mileage or route to that point pair use the Component File to obtain the mileage and route components. This technique saves the computer time and I-O requests required to construct the mileage route a second time.

If a record cannot be added to the Component File due to lack of space, processing will continue normally and the mileage or route will be returned to the user. However, if the same point pair is requested again, the mileage or route must be reconstructed.

In certain cases component records are not written to the file. This is always the case when the origin and destination are HHG keypoints. The mileage can be obtained directly from the Mileage File, so using the Component File would not decrease processing time.

The Component File is most important when the transaction volume is high or requests for a given origin-destination are repeated often. The user must weigh the advantage of time savings against the cost of the extra disk space.

If the user chooses not to use the file, the DD statements for the MI390D dataset should be omitted from the CICS start-up JCL and the batch JCL. The entry for the dataset in the CICS FCT table can also be omitted if desired, but that is not necessary. Also, the Component File flag on the Parameter File MI115 record must be set to 'N'. If the flag is set to 'Y', the system expects the Basic Component File (MI390D) to be present. See the MILEMAKER USER MANUAL for further information concerning the Parameter File update process.

Billing File (MI520D)

The Billing File contains audit trail information for all activity. A record is written to the Billing File for each transaction processed by MileMaker. This includes mileage and route requests, and Geographic File browses. The data stored includes the user id, date, time, terminal id, transaction type, and origin and destination SPLC's. The file is available to track system usage. Separate Billing files must be established for batch and on-line (see the SHAREOPTION information in Chapter 1).

This file is available for the client to track his system usage. If the client chooses not to use the file, the DD statements for the MI520D dataset should be omitted from the CICS start-up JCL and the batch JCL. The dataset entry in the CICS FCT table can also be omitted, but that is not necessary. The Billing File flag on the Parameter File MI115 record must also be set to 'N' to bypass the MI520D file (see MileMaker USER MANUAL).

The Billing File record is described below:

Cobol Name	Length	Description
MI520D-KEY	19	Record key. Fields described below.
MI520D-CLIENT-ID	4	Online-operator id from sign-on. Batch-XXXX or passed client-id.
MI520D-MONTH	2	2-digit month.
MI520D-DAY	2	2-digit day.
MI520D-TIME	4	Time in packed format (HHMMSSs).
MI520D-ORDER	2	Point pair number if multiple points in inquiry (packed).
MI520D-TRANS-TYPE	1	Type of inquiry.
		M=Mileage Inquiry O=Origin Inquiry
		S=Shortest Dist Route D=Batch Short.
		P=Practical Route T=Batch Pract.
		Z=3-digit ZIP Code Inquiry
		2=Secondary File Inquiry/Update
		4=Geo. Browse Mileage/Origin/Route
		5=Geographic File Inquiry/Update
		R=Trans. from Client interface pgm.
MI520D-TERMINAL	4	CICS terminal id. BTCH=Batch.
MI520D-COUNTER	2	Packed. Unused at present. Set to 1.
MI520D-ORIGIN-CODE	1	Keypoint code of origin town.
MI520D-DEST-CODE	1	Keypoint code of destination town.
MI520D-ACTIVITY-TYPE	1	A=Route only on route inquiry
		B=State mileage breakdown (SMB) only
		C=Route with state mileage breakdown
		D=Full route with SMB Shortest only
		H=Practical Miles Only
		1,2,3=Geographic/Secondary File Inquiry/Update/Browse
		Blank = Batch
MI520D-TRANSACTION	4	Transaction id. First byte has G for current guide inquiry, remainder of field unused at present.
MI520D-PGM-MODULE	8	Program which wrote record.
MI520D-LOW-SPLC	5	Origin SPLC (packed).
MI520D-HIGH-SPLC	5	Destination SPLC (packed).
MI520D-YEAR	2	Blanks at present.
MI520D-TASK	4	CICS task number (packed).
FILLER	8	Unused.
Total Length	**60**	

Geographic Update File (GE020D)

The Geographic Update File is required if the user intends to make updates to his Geographic File (GE010D). When a record is added or changed on the GE010D file, the update is recorded on the GE020D file.

This information may be required if a new GE010D file is issued by Rand McNally. Instructions concerning the handling of the client's GE020D file will be provided as needed with a new release. For Release 19 installation the GE020D which exists at the user site is not required. The user should delete his GE020D file and load the file distributed by Rand McNally using the GE020DVD procedure. This file contains only a dummy record.

Note that the software will not allow certain updates to the GE010D file if the GE020D file is not present.

If the client chooses not to use the file, the DD statements for the GE020D dataset should be omitted from the CICS start-up JCL and the batch JCL. The entry for the dataset in the CICS FCT table can also be omitted if desired but that is not necessary. In this case, additions, deletions and certain change requests to the Geographic File will not be allowed by the MileMaker software.

ZIP Mileage File (MI015D) and ZIP Master File (MI105D)

ZIP Code Inquiry is a feature of the mileage inquiry which allows the user to retrieve mileages in accordance with the current Household Goods National 3-digit ZIP Code Mileage Guide.

ZIP Code inquiry is available in the online MileMaker system. Use of this feature is described in the MileMaker User Manual. In the batch environment, ZIP Code Inquiry is available using either the Batch Mileage Inquiry (MI200) or the Batch Interface (MI210). Refer to MileMaker Mileage/ZIP Interface Documentation And The Batch Mileage Inquiry Documentation for details.

If this feature is used, the MI015D and MI105D files are required. In addition, the user id 'ZIP3' must be defined on the Parameter File (PM010D) in order to use this feature in the online environment. Refer to the User Manual for instructions in adding this user id. If the client chooses not to use this feature DD statements for the MI015D and MI105D datasets should be omitted from the CICS start-up JCL and the batch JCL. The entries for the datasets in the CICS FCT table can also be omitted if desired but that is not necessary. To avoid CICS errors, make sure that the 'ZIP3' user id does not exist on the Parameter File if the ZIP Code inquiry is not used.

ZIP5 Reference File (MI882D)

5 Digit ZIP Code inquiry is available in the online MileMaker system. Use of this feature is described in the MileMaker User Manual. In the batch environment, ZIP Code Inquiry is available using either the Batch Mileage Inquiry (MI200) or the Batch Interface (MI210). Refer to MileMaker Mileage/ZIP Interface Documentation And The Batch Mileage Inquiry Documentation for details.

PARAMETER FILE DESCRIPTION

Chapter 5

Chapter Contents

Parameter File Description

The Parameter file (PM010D) contains client user IDs, passwords, and MileMaker system parameters. The data contained in the Parameter file is explained in further detail in the MileMaker User Manual.

The Parameter file data contained on the Distribution tape contains all the system data required to operate MileMaker. The client, however, must add his own user profiles and passwords. Refer to the MileMaker User Manual for more information regarding creation of the user passwords and profiles. To gain initial access to the system, Rand McNally provides the user ID "TDM" and the password "<<<<<<<<" (four less-than signs). This password has full security and update capability.

Rand McNally recommends that the client establish a master profile and password with full update capability. These can subsequently be used to delete the distributed profile and password. The master profile should be restricted to authorized personnel only.

Several flags relating to the use of optional MileMaker features must be set by the user. Refer to the MileMaker User Manual for a description of these flags.

If the client chooses not to use the MileMaker supplied security system, the "Sign-On Flag" on the Parameter file MI115 should be set to "N". This bypasses MileMaker's security checking and will allow full update capability. This now becomes the user's responsibility. If this option is used, do not delete the Rand McNally-provided user ID "RAND" and the password "RAND" from the Parameter file.

BATCH FACILITIES

Chapter 6

Chapter Contents

Batch Mileage Inquiry

The Batch Mileage Inquiry allows the user to obtain mileages on a printed report or in an output tape or disk file. The programs which make up the batch system follow:

Main Program:	MI200		
Subprograms:	GE101	MI220	GE102
	MI201	MI225	
	MI205	MI230	
	MI215	MI240	

The subprograms must be compiled to the relocatable library and the main program must be recompiled to the load library using options: TRUNC(BIN).

Complete instructions for the use of the Batch Inquiry can be found in the document entitled Batch Mileage/ZIP Inquiry documentation.

The JCL used to execute MI200 is included on dataset "TDM.JCL.T" on the distribution tape. The member name is MI200X. The JCL requires modification for optional features and user dataset and library names.

Batch Mileage Interface

The Batch Mileage Interface allows the user to call MileMaker from a user-written program to obtain mileages. The programs which are used in the interface processing follow:

Main Program:	MI200		
Main Program:	User calls MI210		
Subprograms:	GE101	MI215	GE102
	MI201	MI220	
	MI205	MI225	
	MI210	MI240	

The user must compile the MileMaker modules (shown above) into the load library using options: TRUNC(BIN). They must be linked together with the user-written calling program. The user program calls MI210 and processes the response from MI210.

Complete instructions for the use of the Batch Inquiry can be found in the document entitled Batch Mileage/ZIP Inquiry documentation.

Sample JCL containing the Dataset statements required to use the Batch Interface can be found in member MI210X on dataset "TDM.JCL.T" on the System Distribution Tape. These Dataset statements should be modified for user dataset names and added to the JCL for the execution of the user-written program which calls MI210.

MI921X - Key Point and Node Master List

Procedure: MI921X

Description: List the Key Points and Nodes contained on the Geographic file (GE0101D)

Module: MI921

Narrative: This module will produce a report showing all the Key Points and Nodes contained on the Geographic file.

Input Files

DLBL	Type	Description
GE010DP	VSAM	Geographic File

Output Files

DLBL	Type	Description
KEYLIST	RPT	Key Point and Node listing

CPU Requirements

The MI921X procedure should take less than one minute of CPU time to execute.

JCL Requirements

The JCL has been provided in the member named MI921X and contained on the file "TDM.JCL.T" of the System Distribution Tape. Before executing, please modify the JCL to your standards.

Upon completion of execution, the step should return a completion code of 000.

Error Messages

Messages: MI921 ABNORMAL END, GEOGRAPHIC OPEN ERROR, STATUS CODE = xx.

Action: Verify the Geographic File GE010DP. Possible contention with another job; check shareoptions.

Message: MI921 ABNORMAL END, GEOGRAPHIC READ ERROR, STATUS CODE = xx.

Action: Verify the Geographic File GE010DP. If the problem persists, the files may need to be reloaded. The program uses simple sequential access; this error cannot be caused by an invalid key.

MI951X - Geographic File List

Procedure: MI951X

Description: List the locations contained on the Geographic file (GE010DP)

Module: MI951

Narrative: This module will produce a report showing all the location records contained on the Geographic file.

Input Files

DLBL	Type	Description
GE010DP	VSAM	Geographic File

Output Files

DLBL	Type	Description
KEYLIST	RPT	Geographic File listing

CPU Requirements

The MI951X procedure should take less than one minute of CPU time to execute.

JCL Requirements:The execution JCL has been provided in the member named MI951X and contained on the file "TDM.JCL.T" of the System Distribution Tape. Before executing, please modify the JCL to your standards.

Upon completion of execution, the step should return a completion code of 000.

Error Messages

Message: MI951 ABNORMAL END, GEOGRAPHIC OPEN ERROR, STATUS CODE = xx.

Action: Verify the Geographic File GE010DP.

Message: MI951 ABNORMAL END, GEOGRAPHIC READ ERROR, STATUS CODE = xx.

Action: Verify the Geographic File GE010DP. If the problem persists, the files may need to be reloaded. The program uses simple sequential access; this error cannot be caused by an invalid key.

MI9390DX - Component File Delete

The MI9390DX procedure executes program MI9390D.

The JCL for the MI9390DX procedure is included on the System Distribution Tape in dataset "TDM.JCL.T". A complete description of the use of this procedure is outlined in the document entitled Component File Deletion.

SHORTEST DISTANCE ROUTING INSTALLATION

Chapter 7

Chapter Contents

Introduction

MileMaker - Shortest Distance Routing, a Rand McNally & Company system, provides routes which substantiate HHG mileage. Simply by entering an origin and a destination, a user can quickly and accurately audit an HHG mileage.

For the person or department installing this package, it is helpful to have someone who has a good technical background, knowledge of IBM JCL, specific knowledge in using IDCAMS when creating files, and a good understanding on how to bring up a CICS system.

NOTE: Review all JCL and change it according to your own shop standards, especially the attributes used when defining the VSAM clusters. Your standards may differ from Rand McNally's. In all cases Rand McNally attempts to use general parameters that apply at most customer sites.

The operating environment for MileMaker-SHORTEST DISTANCE ROUTING requires the following minimums:

- IBM z/0S V1.7 operating system
- IBM CICS Transaction Server V2.3
- IBM Enterprise COBOL
- Approximately 1.3 gigabytes of disk space (1600 3390 cylinders) for the data files in addition to the space required for basic MileMaker. If previous guide support is required an additional 757 megabytes (1063 cylinders) will be required.

Installation Guide

The purpose of this chapter is to describe the technical aspects of installing Release 19 MileMaker-Shortest Distance Routing. Each section of this chapter deals with a specific installation topic. Also included in Appendix B are lists and worksheets that contain pertinent information about the installation of the system.

Introduction: This section contains general information concerning the use of this guide and the contents of the installation tapes.

Shortest Routing Installation: This section contains a checklist for the complete installation of MileMaker Shortest Distance Routing.

Special File Installation Procedures: This section gives detailed instruction for creating and/or loading the Secondary Keypoint Route File.

Optional Batch Processing: This section contains information concerning the use of the Batch Shortest Distance Route Display, Batch Route Interface, and Batch Interface with Detailed Routes.

Appendix B: A "File Data Set Name Worksheet" has been provided to help keep a record of data set names that you will change from Rand McNally's naming conventions to yours. We suggest that you complete this form prior to modifying any JCL. Also, you may modify any of the IDCAMS options to suit your own standards.

Installation Tapes

Depending on which package has been purchased, two or more tapes will be distributed for the initial release of MileMaker Shortest Distance system. The cover letter that accompanied this document will identify which tape(s) contains JCL, tables, load and source modules (System Distribution Tape) and which tape(s) contains the MileMaker data files (Data Distribution Tape).

The "System Distribution Tape" contains PDS files that were created using the IBM utility "IEBCOPY", therefore the same utility must be used to copy them from Rand McNally's tape to your DASD device. This tape contains the following:

Label	DSN	System Distribution Tape Contents
01	TDM.JCL.T	This PDS contains all JCL required in the installation and operation of the system. See Appendix B for a list of members.
02	TDM.TBL.T	This PDS contains all the CICS Table entries required for the CICS system. See Appendix B for list of members.
03	TDM.LDL.T	This PDS contains all the load modules for the CICS and BATCH system(s). See Appendices B for a list of members.
04	TDM.SRC.T	This PDS contains all the source modules for the CICS and/or BATCH system(s). See Appendices A-1and A-2 for a list of members.

Minimum Space And DCB Info For PDS

DSN	I/O	DSORG	RECFM	LRECL	BLKSIZE	ALLOC	DIRECTORY
TDM.JCL.T	I	PO	VS	18108	18112	3 TRKS	20
	O	FB	80	9040			
TDM.TBL.T	I	PO	VS	18108	18112	2 TRKS	20
	O	FB	80	9040			
TDM.LDL.T	I	PO	VS	19085	19089	32 TRKS	50
	O	U	*	19069			
TDM.SRC.T	I	PO	VS	18108	18112	450 TRKS	50
	O	FB	80	9040			

* no record size is specified if RECFM is U

When loading down Rand McNally's system tapes it is important to set up your JCL correctly to ensure that you load down all the data on the tape. You must use the IBM utility IEBCOPY to properly load the data. The example below shows how to set up your JCL. The space requirements in this section are for a 3390 disk.

```
//TDMDATA JOB
//*************************************************************
//** STEP01 COPY PDS TO YOUR SYSTEM
//*************************************************************
//STEP01   EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//JCLIN    DD DSN=TDM.JCL.T, -TDM JCL MEMBERS
//         DISP=(OLD,KEEP,KEEP),
//         UNIT=TAPE,
//         LABEL=1,
//         VOL=SER=000000  -CHANGE TO PROPER VOLSER
//TBLIN    DD DSN=TDM.TBL.T, -TDM TABLE MEMBERS
//         DISP=(OLD,KEEP,KEEP),
//         UNIT=TAPE,
//         LABEL=2,
//         VOL=SER=000000  -CHANGE TO PROPER VOL SER
//LDLIN    DD DSN=TDM.LDL.T, -TDM LOAD MEMBERS
//         DISP=(OLD,KEEP,KEEP),
//         UNIT=TAPE,
//         LABEL=3,
//         VOL=SER=000000  -CHANGE TO PROPER VOL SER
//SRCIN    DD DSN=TDM.SRC.T, -TDM SOURCE MEMBERS
//         DISP=(OLD,KEEP,KEEP), -(OPTIONAL)
//         UNIT=TAPE,
//         LABEL=4,
//         VOL=SER=000000  -CHANGE TO PROPER VOL SER
//JCLOUT   DD DSN=(YOUR JCL LIBRARY NAME),
//         DISP=(NEW,CATLG,DELETE),
//         UNIT=SYSDA,
/          SPACE=(TRK,(3,,20),RLSE),
//         DCB=(RECFM=FB,LRECL=80,BLKSIZE=9040,DSORG=PO)
//TBLOUT   DD DSN=(YOUR TABLE LIBRARY NAME),
//         DISP=(NEW,CATLG,DELETE),
//         UNIT=SYSDA,
//         SPACE=(TRK,(2,,20),RLSE),
//         DCB=(RECFM=FB,LRECL=80,BLKSIZE=9040,DSORG=PO)
//LDLOUT   DD DSN=(YOUR LOAD LIBRARY NAME),
//         DISP=(NEW,CATLG,DELETE),
//         UNIT=SYSDA,
//         SPACE=(TRK,(90,,25),RLSE),
//         DCB=(RECFM=U,BLKSIZE=19069,DSORG=PO)
```

```
//SRCOUT   DD DSN=(YOUR SOURCE LIBRARY NAME),
//         DISP=(NEW,CATLG,DELETE),
//         UNIT=SYSDA,
//         SPACE=(TRK,(450,,50),RLSE),
//         DCB=(RECFM=FB,LRECL=80,BLKSIZE=9040,DSORG=PO)
//SYSUT3   DD UNIT=SYSDA,SPACE=(TRK,(1))
//SYSUT4   DD UNIT=SYSDA,SPACE=(TRK,(1))
//SYSIN    DD *
  COPY OUTDD=JCLOUT,INDD=JCLIN
  COPY OUTDD=TBLOUT,INDD=TBLIN
  COPY OUTDD=LDLOUT,INDD=LDLIN
  COPY OUTDD=SRCOUT,INDD=SRCIN
/*
//
```

The "Data Distribution Tape" contains data files for the MileMaker-Shortest Distance Routing system. These tapes contain the files listed below. These files are sequential in format. Refer to the distribution cover letter for actual tape file Label assignments and tape volume serial numbers.

Data Distribution Tape

DSN	RECFM	LRECL	BLKSIZE	CONTENTS
TDM.MI320D.T	VB	1092	32260	Highway blocker file.
TDM.MI340R.T	VB	2322	32558	Shortest distance route (A-K-B) file.
TDM.MI605R.T	VB	208	32000	Secondary key point route file.

Note: for all files with RECFM=VB, LRECL=MAXIMUM LRECL in IDCAMS define + 4 byte LLBB necessary for tape

Shortest Routing Installation

Prior to loading the Shortest Distance Routing programs and files, Basic MileMaker must be installed. The route processing uses the Basic MileMaker programs and files in determining the route. Follow the instructions in the Basic MileMaker Initial Installation Guide. If the client received Basic MileMaker and Shortest Distance Routing in the same distribution, then the programs, JCL, and tables for both systems are contained on the same System Distribution Tape. In this case the programs, JCL and tables are loaded to your source library when Basic MileMaker is loaded. You should begin the Shortest Distance Routing installation with Step 02 of the checklist.

The JCL referred to in the checklist is contained on the System Distribution Tape. The file name is "TDM.JCL.T". The PDS member name is the same as the

procedure name. Before executing these procedures, they should be reviewed. The client may wish to change the dataset names or other parameters to conform to the client's installation standards. The client must also enter the input tape volume serial numbers and the disk volumes on which the datasets will reside.

The files are included on the Data Distribution Tape(s). The volume serial numbers and contents of the tapes are included in the Installation Cover Letter. The file allocation JCL procedures supplied for Shortest Distance Routing all contain three logical steps except for the procedures described in the **Special File Installation Procedures** section. These are a file delete, file allocation, and file load. Upon completion of execution, the delete step returns a completion code of 08 if the dataset did not already exist. If the dataset did already exist, the dataset is deleted and a completion code of zero is returned. The file allocation and load operations should always return a completion code of zero.

The modules referred to in Steps 6 and 7 of the installation checklist are contained in the System Distribution Tape on dataset "TDM.SRC.T". A list of the programs required for Shortest Distance Routing can be found in Appendix A-1 and A-2. After the System Distribution Tape is loaded onto your system (see

Introduction above), the load modules should be moved to the CICS or Batch Application Load Library. The source code is also supplied in case recompilation is required. The source code can be found on the System Distribution Tape on dataset "TDM.SRC.T". These members are not needed unless recompilation is required. If recompilation is required, then one of the following compiler options must be used:

```
CICS COBOL OS/390: RENT, NODYNAM, TRUNC(BIN), NOOPT, DATA(31)
BATCH COBOL OS/390: RENT, DYNAM, TRUNC(BIN), NOADV, NOOPT,
DATA(31)
```

Guide 19 Installation Checklist

Use the following checklist to install Shortest Route. It is assumed that all Guide 19 files and modules are initially placed into a test environment for verification.

- ☐ **STEP 1** Unload SYSTEM DISTRIBUTION TAPE or CD-ROM.
- ☐ Unload "TDM.JCL.T" from SYSTEM DISTRIBUTION TAPE.
- ☐ Unload "TDM.TBL.T" From System Distribution Tape.
- ☐ Unload "TDM.LDL.T" From System Distribution Tape.
- ☐ Unload "TDM.SRC.T" From System Distribution Tape.
- ☐ **STEP 2** Build And Load Highway Blocker File.
- ☐ Modify MI320DVD Procedure To Your Standards.
- ☐ Execute MI320DVD Procedure.
- ☐ Check Return Codes.
- ☐ **STEP 3** Build And Load Shortest Distance Route (A-K-B) File.
- ☐ Modify MI340RVD Procedure To Your Standards.
- ☐ Execute MI340RVD Procedure.
- ☐ Check Return Codes.
- ☐ **STEP 4** Build And Load Secondary Keypoint Route Files.
  See **Special File Installation Procedures** below.
- ☐ **STEP 5** CICS Tables.
- ☐ Modify MRDDS To Standards.
- ☐ Add MRDDS Statements To Cics Start-up JCL.
- ☐ Add MRFCT Statements To FCT Table.
- ☐ Add MRPCT Statements To PCT Table.
- ☐ Add MRPPT Statements To PPT TABLE.
- ☐ Assemble FCT Table.
- ☐ Assemble PCT Table.

☐ Assemble PPT Table.

☐ **STEP 6** CICS Modules.

☐ Load CICS Modules to CICS Application Library.

☐ **STEP 7** Batch Modules (optional) Description of Batch Modules is found in **Optional Batch Processing** below.

☐ Load Batch Modules to Application Library.

☐ **STEP 8** Batch JCL (Optional) Description of Batch Modules is found in **Optional Batch Processing** below.

☐ Load Batch JCL To Application Library.

☐ Modify MI410X Procedure To Conform To Your Standards.

☐ Execute MI410X Procedure.

☐ Check Return Codes And Reports.

☐ Modify MI435X DD Statements To Conform To Your Standards.

☐ Add MI435X Statements To Procedure For User Calling Program.

☐ Execute User Procedure.

☐ Check Return Codes And Reports.

☐ Modify MI7435X Procedure to Conform to your Standards.

☐ Add DD Statements to the procedure for the User Calling Program.

☐ Execute the User Procedure

☐ Check the Return Codes and Reports.

Previous Guide 18 Installation Checklist

In general, Production G18 files will be copied to newly defined G19 previous Guide files. The following steps should be followed ONLY if previous guide Shortest Distance Routing is supported.

☐ **STEP 1** Copy the Production G18 MI320D file to the newly defined G19 MI322D file.

☐ Modify MI322DVD Procedure To Conform To Your Standards.

☐ Execute MI322DVD Procedure.

☐ Check Return Codes.

☐ **STEP 2** Copy the Production G18 MI340R file to the newly defined G19 MI342R file.

☐ Modify MI342RVD Procedure To Conform To Your Standards.

☐ Execute MI342RVD Procedure.

☐ Check Return Codes.

☐ **STEP 3** Build And Load Secondary Keypoint Route File.
See **Special File Installation Procedures** below.

Special File Installation Procedures

MI605R - Secondary Keypoint Route File.

The current guide Secondary Keypoint Mileage File, MI605M, and the Secondary Keypoint Route File, MI605R, are companion files and are required for Shortest Routing. The same MI605R file is used for both Shortest and Practical Routing.

If the MI605R file was already installed for Practical Routing or as a part of the Basic MileMaker installation, then this step can be skipped.

Two procedures are provided in Shortest Distance Routing in order to load the MI605R file. New clients must use procedure 1.

Existing clients should use procedure 1 only if users have not added their own secondary records. If users have added secondary records, procedure 2 must be used.

1. The Rand McNally distributed current guide Secondary Keypoint Route File will be loaded directly from the Data Distribution Tape. Please follow these steps:

   ☐ Modify the MI605RVD Procedure to Conform to Your Standards.

   ☐ Execute MI605RVD Procedure.

   ☐ Check Return Codes (See Below).

   ☐ Modify MI605RVR Procedure to Conform to Your Standards.

   ☐ Execute MI605RVR Procedure.

   ☐ Check Return Codes. (A single MI605R dummy record will be loaded)

   If return codes are all zero, continue with the next step in the installation checklist in the section **Shortest Routing Installation**.

2. Existing clients must run an edit/update procedure, MI9605X, against their current Guide secondary files to insure that the geographic related fields are in sync with the Geographic File, GE010D. This procedure should have been executed as part of the Basic MileMaker installation.

   Refer to Chapter 3 for further information. The steps needed to load the MI605R file are listed there.

MI607R - Previous Guide Secondary Keypoint Route File

The previous guide Secondary Keypoint Mileage File, MI607M, and the Secondary Keypoint Route File, MI607R, are companion files and are required for Shortest Distance Routing.

If the MI607R file was already installed as a part of the Basic MileMaker installation, then this step can be skipped.

New customers will skip this procedure since previous Guide 18 files are not distributed in the Guide 19 update.

Existing customers can skip this step by deactivating previous Guide files from the parameter update option from the CICS MileMaker main menu.

Your Production Guide 18 MI605R file will be copied to a newly defined Guide 19 MI607R. Please follow these steps:

- ☐ Modify the MI607RVD Procedure to Conform to Your Standards.
- ☐ Execute MI607RVD Procedure.
- ☐ Check Return Codes (See Below).
- ☐ Modify MI607RVR Procedure to Conform to Your Standards.
- ☐ Execute MI607RVR Procedure.
- ☐ Check Return Codes.

If return codes are all zero, continue with the next step in the installation checklist in the section on **Shortest Routing Installation**.

Optional Batch Processing

MI410X - Batch Route Report

The Batch Reporting option allows the client to receive a printed report containing routes and/or state mileage data.

Program MI410 is provided on the System Distribution Tape. It is contained in dataset "TDM.LDL.T". See the **Installation Tapes** section of this chapter for further information concerning the System Distribution Tape. Sample JCL for the execution of MI410 is contained in dataset "TDM.JCL.T", also on the System Distribution Tape. The member name is MI410X.

For a detailed description of the use of the batch Shortest Route report option, see the enclosed documentation titled Batch Route Inquiry Documentation.

The programs which make up the batch route system follow:

Main Program:	MI410	
Subprograms:	GE101	MI225 GE102
	MI201	MI240
	MI215	MI420
	MI220	MI430

The MI410 load module on the Distribution tape was compiled under Enterprise COBOL and dynamically calls its subprograms and COBOL system routines.

MI435X - Batch Route Interface

The batch interface option allows the client to receive state mileage data in table form by calling MileMaker from a user program.

Program MI435 is provided on the System Distribution Tape. It is contained in dataset "TDM.LDL.T". See the **Installation Tapes** section of this chapter for further information concerning the System Distribution Tape. Sample JCL for the execution of MI435 is contained in dataset 'TDM.JCL.T', also on the System Distribution Tape. The member name is MI435X.

For a detailed description of the use of the batch Shortest Route Interface (and Working Sample Program - MI9435), see the enclosed documentation titled Practical And Shortest Distance Routing Interface Documentation.

The programs which make up the route interface follow:

Main Program:	User-written program	
Subprograms:	GE101	MI225 GE102
	MI201	MI240
	MI215	MI420
	MI220	MI435

The MI435 load module on the Distribution tape was compiled under Enterprise COBOL and dynamically calls its subprograms and COBOL system routines. The following compile options must be selected:

- Enterprise COBOL user program calls Enterprise COBOL MI435.

  RENT, DYNAM, TRUNC (BIN), NOADV, NOOPT

MI7435X - Batch Route Interface with Routes

The Batch Route Interface option allows the user to retrieve state mileage and detailed route data in table form by calling MileMaker from a user program.

The program MI7435 is provided on the System Distribution tape. It is contained in dataset "TDM.LDL.T" .(See the **Installation Tapes** section of this chapter for more information concerning the Data Distribution tape.) Sample JCL for the execution of MI7435 is contained in dataset "TDM.JCL.T" of the System Distribution tape; the member name is MI79435X.

For a detailed description of the utilization of the Batch Route Interface with Routes (and working sample program MI79435) see Chapter 12 - Practical and Shortest Routing Interface.

The programs which make up the route interface follow:

Main Program:	User-written program	
Subprograms:	GE101	MI225 GE102
	MI201	MI240
	MI215	MI7420
	MI220	MI7435

The MI7435 load module on the Distribution tape was compiled under Enterprise COBOL and dynamically calls its subprograms and COBOL system routines. The following compile options must be selected:

- Enterprise COBOL user program calls Enterprise COBOL MI7435.

  RENT, DYNAM, TRUNC (BIN), NOADV, NOOPT

PRACTICAL ROUTE INSTALLATION

Chapter 8

Chapter Contents

Introduction

For the person or department installing this package, it is helpful to have someone who has a good technical background, knowledge of IBM JCL, specific knowledge in using IDCAMS when creating files, and a good understanding on how to bring up a CICS system.

Review all JCL and change it according to your own standards, especially with the attributes when defining the VSAM clusters. Your standards may differ from Rand McNally's. In all cases Rand McNally attempts to use general parameters that apply at most customer sites.

The operating environment for MILEMAKER-PRACTICAL ROUTING requires the following minimums:

- IBM z/0S V1.7 operating system
- IBM CICS Transaction Server V2.3
- IBM Enterprise COBOL
- Approximately 745 megabytes of disk storage (920 cylinders on a 3390 device) are needed to load required Practical files. This is in addition to space required for Basic MileMaker files needed for Practical routing. See Appendix B for detailed information concerning file types and sizes.

Initial Installation Guide

The purpose of this chapter is to describe the technical aspects of installing the MILEMAKER-PRACTICAL ROUTING system. Each section of this chapter deals with a specific installation topic. Also included in Appendix C are lists and worksheets that contain pertinent information about the installation of the system.

Introduction: This section contains general information concerning the use of this guide and the contents of the installation tapes.

Practical Route Installation: This section contains a checklist for the complete installation of MileMaker Practical Routing.

Special File Installation Procedures: This section gives detailed instructions for creating and loading the Secondary Keypoint Route File.

Optional File Description: This section discusses the use of the Practical Component File.

Optional Batch Processing: This section contains information concerning the use of the Batch Practical Route Display, Batch Interface Module, and Batch Interface with Detail Routes module.

Appendix C: The "File Data Set Name Worksheet" has been provided to help keep a record of data set names that you will change from Rand McNally's naming conventions to yours. We suggest that you complete this form prior to modifying any JCL. Also, you may modify any of the IDCAMS options to suit your own standards.

Installation Tapes

There are several tapes for the initial installation of the MILEMAKER-PRACTICAL ROUTING system. The cover letter that accompanies this document identifies which tape is the "SYSTEM DISTRIBUTION TAPE" and which are the "DATA DISTRIBUTION TAPES".

The "SYSTEM DISTRIBUTION TAPE" contains PDS files that were created using the IBM utility "IEBCOPY", therefore the same utility must be used to copy them from Rand McNally's tape to your DASD device. This tape contains the following:

Label	DSN	System Distribution Tape Contents
01	TDM.JCL.T	This PDS contains all JCL required in the installation and operation of the system. See Appendix A-6 for a list of members.
02	TDM.TBL.T	This PDS contains all the CICS Table entries required for the CICS system. See Appendix A-5 for list of members.
03	TDM.LDL.T	This PDS contains all the load modules for the CICS and BATCH system(s). See Appendices A-1and A-2 for a list of members
04	TDM.SRC.T	This PDS contains all the source modules for the CICS and/or BATCH system(s). See Appendices A-1and A-2 for a list of members.

Minimum Space And DCB Info For PDS

DSN	I/O	DSORG	RECFM	LRECL	BLKSIZE	ALLOC	DIRECTORY
TDM.JCL.T	I	PO	VS	18108	18112	3TRKS	20
	O		FB	80	9040		
TDM.TBL.T	I	PO	VS	18108	18112	2TRKS	20
	O		FB	80	9040		
TDM.LDL.T	I	PO	VS	19085	19089	32TRKS	50
	O		U	*	19069		
TDM.SRC.T	I	PO	VS	18108	18112	450TRKS	50
	O		FB	80	9040		

* no record size is specified if RECFM is U.

When loading down Rand McNally's system tapes, it is important to set up your JCL correctly to ensure that you load down all the data on the tape. You must use

the IBM utility of IEBCOPY to properly load down the data. The example below shows how to set up the JCL. The space requirements in this example are for a 3380 disk.

```
//TDMSYST JOB --YOUR JOB CARD
//*************************************************************
//STEP01 COPY PDS TO YOUR SYSTEM
//*************************************************************
//STEP01   EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//JCLIN DD DSN=TDM.JCL.T, -TDM JCL MEMBERS
//         DISP=(OLD,KEEP,KEEP),
//         UNIT=TAPE,
//         LABEL=1,
//         VOL=SER=000000
//TBLIN    DD DSN=TDM.TBL.T, -TDM SOURCE MEMBERS
//         DISP=(OLD,KEEP,KEEP),
//         UNIT=TAPE,
//         LABEL=2,
//         VOL=SER=000000
//LDLIN    DD DSN=TDM.LDL.T, -TDM LOAD MEMBERS
//         DISP=(OLD,KEEP,KEEP),
//         UNIT=TAPE,
//         LABEL=3,
//         VOL=SER=000000
//SRCIN    DD DSN=TDM.SRC.T, -TDM SOURCE MEMBERS
//         DISP=(OLD,KEEP,KEEP),
//         UNIT=TAPE,
//         LABEL=4,
//         VOL=SER=000000
//JCLOUT   DD DSN=(YOUR JCL LIBRARY NAME),
//         DISP=(NEW,CATLG,DELETE),
//         UNIT=SYSDA,
//         SPACE=(TRK,(10,,20),RLSE),
//         DCB=(RECFM=FB,LRECL=80,BLKSIZE=9040,DSORG=PO)
//TBLOUT   DD DSN=(YOUR TABLE LIBRARY NAME),
//         DISP=(NEW,CATLG,DELETE),
//         UNIT=SYSDA,
//         SPACE=(TRK,(10,,20),RLSE),
//         DCB=(RECFM=FB,LRECL=80,BLKSIZE=9040,DSORG=PO)
//LDLOUT   DD DSN=(YOUR LOAD LIBRARY NAME),
//         DISP=(NEW,CATLG,DELETE),
//         UNIT=SYSDA,
//         SPACE=(TRK,(90,,25),RLSE),
//         DCB=(RECFM=U,BLKSIZE=19069,DSORG=PO)
```

```
//SRCOUT   DD DSN=(YOUR SOURCE LIBRARY NAME),
//         DISP=(NEW,CATLG,DELETE),
//         UNIT=SYSDA,
//         SPACE=(TRK,(450,,50),RLSE),
//         DCB=(RECFM=FB,LRECL=80,BLKSIZE=9040,DSORG=PO)
//SYSUT3   DD UNIT=SYSDA,SPACE=(TRK,(1))
//SYSUT4   DD UNIT=SYSDA,SPACE=(TRK,(1))
//SYSIN    DD *
  COPY OUTDD=JCLOUT,INDD=JCLIN
  COPY OUTDD=TBLOUT,INDD=TBLIN
  COPY OUTDD=LDLOUT,INDD=LDLIN
  COPY OUTDD=SRCOUT,INDD=SRCIN
/*
//
```

The "DATA DISTRIBUTION TAPES" contain data files for the MileMaker PRACTICAL ROUTING system. These tapes contain the files listed below. Refer to the distribution cover letter for actual tape file LABEL assignments.

Data Distribution Tape

DSN	RECFM	LRECL	BLKSIZE	CONTENTS
TDM.MI321D.T	VB	1092	31204	Highway blocker file.
TDM.MI341R.T	VB	1098	31204	Practical distance route (A-K-B) file.
TDM.MI306D.T	VB	1811	32204	Surrounding blocker file.
TDM.MI021D.T	FB	2036	30540	Practical locator file.
TDM.MI351D.T	FB	9	27000	Keypoint cross reference file.
TDM.MI391D.T	VB	46	51	Optional component file.
TDM.MI141B.T	FB	4088	28616	Time/route file.
TDM.MI605R.T	VB	208	32000	Secondary keypoint route file.

Note: for all files with RECFM=VB, LRECL=MAXIMUM LRECL in IDCAMS define + 4 byte LLBB necessary for tape.

Practical Route Installation

Prior to loading the Practical Routing programs and files, Basic MileMaker must be installed. If you are using only MileMaker Practical Routing, then perform only those steps indicated in the Basic installation. The route processing uses the Basic MileMaker programs and files in determining the route. If Basic MileMaker has not been installed, follow the instructions in the Basic Milemaker Initial Installation Guide. If the client received Basic MileMaker and Practical Routing in the same

distribution, then the programs, JCL, and tables for both systems are contained on the same System Distribution Tape. In this case, the programs, JCL and tables are loaded to your source library when Basic MileMaker is loaded. You should begin the Practical Routing installation with Step 02 in the

Installation Checklist section below.

The JCL referred to in the checklist is contained on the System Distribution Tape. The file name is "TDM.JCL.T". Before executing these procedures, they should be reviewed. The client may wish to change the dataset names or other parameters to conform to the client's installation standards. The client must also enter the volume serial numbers of the Data Distribution Tapes and the disk volumes on which the datasets will reside.

The files are included on the Data Distribution Tapes. The volume serial numbers and contents of the tapes are included in the Installation Cover Letter. The file allocation JCL procedures supplied for Practical Routing all contain three logical steps. These are a file delete, file allocation, and file load. Upon completion of execution, the delete step returns a completion code of 08 if the dataset did not already exist. If the dataset did already exist, the dataset is deleted and a completion code of zero is returned. The file allocation and load operations should always return a completion code of zero.

The modules referred to in Steps 11 and 12 of the installation checklist are contained in the System Distribution Tape on dataset "TDM.LDL.T". A list of the programs required for Practical Routing can be found in Appendices A-1 and A-2. After the System Distribution Tape is loaded onto your system (see the Introduction above), these load modules should be moved to the CICS or Batch Application Load Library.

The source code is also supplied in case recompilation is required. The source code can be found on the System Distribution Tape on dataset "TDM.SRC.T". These members are not needed unless recompilation is required. If recompilation is required, then one of the following compiler options must be used:

```
CICS COBOL OS/390: RENT, NODYNAM, TRUNC(BIN), NOOPT, DATA(31)
BATCH COBOL OS/390: RENT, DYNAM, TRUNC(BIN), NOADV, NOOPT,
DATA(31)
```

Installation Checklist

Use the following checklist to install Practical Route

- ☐ **STEP-01** Unload System Distribution Tape or CD-ROM (see Note Above).
- ☐ Unload "TDM.JCL.T" From System Distribution Tape.
- ☐ Unload "TDM.TBL.T" From System Distribution Tape.
- ☐ Unload "TDM.LDL.T" From System Distribution Tape.
- ☐ Unload "TDM.SRC.T" From System Distribution Tape.

- ☐ **STEP-02:** Build and Load Highway Blocker File.
- ☐ Modify MI321DVD PRocedure to Your Standards.
- ☐ Execute MI321DVD Procedure.
- ☐ Check Return Codes.

- ☐ **STEP-03:** Build and Load Practical Distance Route (A-K-B) File.
- ☐ Modify MI341RVD Procedure to Your Standards.
- ☐ Execute MI341RVD Procedure.
- ☐ Check Return Codes.

- ☐ **STEP 04:** Build and Load Time/route File.
- ☐ Modify MI141BVD Procedure to Your Standards.
- ☐ Execute MI141BVD Procedure.
- ☐ Check Return Codes.

- ☐ **STEP-05:** Build And Load Locator File.
- ☐ Modify MI021DVD Procedure to Your Standards.
- ☐ Execute MI021DVD Procedure.
- ☐ Check Return Codes.

- ☐ **STEP-06:** Build And Load Blocker File.
- ☐ Modify MI306DVD Procedure to Your Standards.
- ☐ Execute MI306DVD Procedure.
- ☐ Check Return Codes.

- ☐ **STEP-07:** Build and Load Cross-reference File.
- ☐ Modify MI351DVD Procedure to Your Standards.

☐ Execute MI351DVD Procedure.

☐ Check Return Codes.

☐ **STEP-08:** Build and Load Secondary Keypoint Route File.
See the **Special File Installation Procedures** section of this chapter.

☐ **STEP-09:** Build and Load Route Component File. (optional) **

☐ Modify MI391DVD Procedure to Your Standards.

☐ Execute MI391DVD Procedure.

☐ Check Return Codes.

☐ **STEP-10:** CICS Tables.

☐ Modify PRDDS DD Statements to Conform to Your Standards

☐ Add PRDDS Statements to CICS START-UP JCL.

☐ Add PRFCT Statements To FCT Table.

☐ Add PRPCT Statements To PCT Table.

☐ Add PRPPT Statements To PPT Table.

☐ Assemble FCT Table.

☐ Assemble PCT Table.

☐ Assemble PPT Table.

☐ **STEP-11:** CICS Modules.

☐ Load CICS Modules to CICS Application Library.

☐ **STEP-12:** Batch Modules (optional): ***

☐ Load Batch Modules to Application Library.

☐ **STEP-13:** Batch JCL (optional):

☐ Load Batch JCL to Application Library.

☐ Modify MI410X Procedure to Conform to Your standards.

☐ Execute MI410X PROCEDURE.

☐ Check Return Codes And Reports.

☐ Modify MI435X Procedure to Conform to Your standards.

☐ Add MI435X Statements to Calling Program procedure.

☐ Execute User Procedure.

☐ Check Return Codes and Reports.

- ☐ Modify MI7435X Procedure to Conform to Your standards.
- ☐ Add MI7435X Statements to Calling Program procedure.
- ☐ Execute User Procedure.
- ☐ Check Return Codes and Reports.

The **Special File Installation Procedures section below contains descriptions of optional files.

***The **Optional Batch Processing** section below contains descriptions of optional batch modules.

Special File Installation Procedures

MI605R - Secondary Keypoint Route File.

The current guide Secondary Keypoint Mileage File, MI605M, and the Secondary Keypoint Route File, MI605R, are companion files and are required for Practical Routing. The same MI605R file is used for both Shortest and Practical Routing.

If the MI605R file was already installed for Shortest Distance Routing or as a part of the Basic MileMaker installation, then this step can be skipped.

No new Secondary Route records are distributed in the Guide 19 update.

Two procedures are provided in Practical Routing in order to load the MI605R file. New clients must use procedure 1.

Existing clients should use procedure 1 only if users have not added their own secondary records. If users have added secondary records, procedure 2 must be used.

1. The Rand McNally distributed current guide Secondary Keypoint Route File will be loaded directly from the Data Distribution Tape. Please follow these steps.

- ☐ Modify the MI605RVD Procedure to Conform to Your Standards.
- ☐ Execute MI605RVD Procedure.
- ☐ Check Return Codes (See Below).
- ☐ Modify MI605RVR Procedure to Conform to Your Standards.
- ☐ Execute MI605RVR Procedure.
- ☐ Check Return Codes. (A single MI605R dummy record will be loaded)

If return codes are all zero, continue with the next step in the installation checklist in the **Practical Route Installation** section.

2. Existing clients must run an edit/update procedure, MI9605X, against their current Guide secondary files to insure that the geographic related fields are in sync with the Geographic File, GE010D. This procedure should have been executed as part of the Basic MileMaker installation.

   Refer to Chapter 3 for further information. The steps needed to load the MI605R file are listed there.

Optional File Description

Component File (MI391D)

The Component File contains prestored route components. When a Practical Route is requested for the first time for a pair of points, a component record is added to the file. Subsequent requests for a route between that point pair use the Component File to obtain the route components. This technique saves computer time and I-O requests required to construct the route a second time.

If a record cannot be added to the Component File due to lack of space, processing will continue normally and the route will be returned to the user. However, if the same point pair is requested again, the route must be reconstructed.

In certain cases component records are not written to the file. This is always the case when the origin and destination are keypoints. The route components in this case are the origin and destination, so using the Component File would not improve processing efficiency.

The Component File is most important when the transaction volume is high or requests for a given origin-destination are repeated often. The user must weigh the advantage of time savings against the cost of the extra disk space.

If the user chooses not to use the file, the DD/DLBL statements for the MI391D dataset should be omitted from the CICS start-up JCL and the batch JCL. The entry for the dataset in the CICS FCT table can also be omitted if desired, but that is not necessary. Also, the Component File flag on the Parameter File MI115 record must be set to 'N'. If the flag is set to 'Y' the system expects the Practical Component File to be present. See the Milemaker User Manual for further information concerning the Parameter File update process.

Optional Batch Processing

MI410X - Batch Route Report

The Batch Reporting option allows the client to receive a printed report containing routes and/or state mileage data.

Program MI410 is provided on the System Distribution Tape. It is contained in dataset "TDM.LDL.T". See the **Installation Tapes** section of this chapter for further information concerning the System Distribution Tape. Sample JCL for the execution of MI410 is contained in dataset "TDM.JCL.T", also on the System Distribution Tape. The member name is MI410X.

For a detailed description of the use of the batch Practical Route report option, see the enclosed documentation titled Batch Route Inquiry Documentation.

The programs which make up the batch route system follow:

Main Program:	MI410	
Subprograms:	GE101	MI225 GE102
	MI201	MI240
	MI215	MI420
	MI220	MI430

The MI410 load module on the Distribution tape was compiled under Enterprise COBOL and dynamically calls its subprograms and COBOL system routines.

MI435X - Batch Route Interface

The batch interface option allows the client to receive state mileage data in table form by calling MileMaker from a user program.

Program MI435 is provided on the System Distribution Tape. It is contained in dataset "TDM.LDL.T". See **Installation Tapes** section of this chapter for further information concerning the System Distribution Tape. Sample JCL for the execution of MI435 is contained in dataset 'TDM.JCL.T', also on the System Distribution Tape. The member name is MI435X.

For a detailed description of the use of the batch Practical Route Interface (and Working Sample Program - MI9435), see the enclosed documentation titled Practical And Shortest Distance Routing Interface Documentation.

The programs which make up the route interface follow:

Main Program:	User-written program	
Subprograms:	GE101	MI225 GE102

MI201	MI240
MI215	MI420
MI220	MI435

The MI435 load module on the Distribution tape was compiled under Enterprose COBOL and dynamically calls its subprograms and COBOL system routines. The following compile options must be selected:

- Enterprise COBOL user program calls Enterprise COBOL MI435.

  RENT, DYNAM, TRUNC(BIN), NOADV, NOOPT

MI7435X - Batch Route Interface with Routes

The Batch Route Interface option allows the user to retrieve state mileage and detailed route data in table form by calling MileMaker from a user program.

The program MI7435 is provided on the Systen Distribution tape. It is contained in dataset "TDM.LDL.T" .(See **Installation Tapes** section of this chapter for more information concerning the Data Distribution tape.) Sample JCL for the execution of MI7435 is contained in dataset "TDM.JCL.T" of the System Distribution tape; the member name is MI7435X.

For a detailed description of the utilization of the Batch Route Interface with Routes (and working sample program MI79435) see Chapter 12 - Practical and Shortest Routing Interface.

The programs which make up the route interface follow:

Main Program:	User-written program	
Subprograms:	GE101	MI225 GE102
	MI215	MI7420
	MI215	MI7420
	MI220	MI7435

The MI7435 load module on the Distribution tape was compiled under Enterprise COBOL and dynamically calls its subprograms and COBOL system routines. The following compile options must be selected:

- Enterprise COBOL user program calls Enterprise COBOL MI435.

  RENT, DYNAM, TRUNC(BIN), NOADV, NOOPT

BATCH MILEAGE/ZIP INQUIRIES

Chapter 9

Chapter Contents

Introduction

The mileage inquiry program reads input containing points for mileage or ZIP Code inquiries. The data is edited and an error report is produced for points in error. For valid points, the mileage is obtained and printed. An output file is also produced containing all points on the input along with the appropriate mileages.

The input data may be in either of two formats. Format 1, multiple record input, consists of an origin record followed by one or more destination records. Optionally, additional sets of origin/ destination records may be included. Format 2 input, single record, consists of a series of records each containing an origin and a destination.

Format 1 input is card image input; that is, the record length is 80. Format 2 input records are 116 characters long.

All city/states and SPLC's (origins and destinations) are checked to ensure that they are on the Geographic File. ZIP Codes are checked against the ZIP Code Master File. All points in error are listed on the Error Report. In the case of multiple record format, if an origin is in error, all of the associated destinations will appear on the report.

The programs which make up the batch system follow:

Main Program:	MI200	
Subprograms:	GE101	MI220 GE102
	MI201	MI225
	MI205	MI230
	MI215	MI240

The MI200 load module on the Distribution tape was compiled under Enterprise COBOL and dynamically calls its subprograms and COBOL system routines.

The JCL used to execute MI200 is included on dataset 'TDM.JCL.T' on the distribution tape. The member name is MI200X. The JCL requires modification for optional features and user dataset and library names.

Parameter Description

A parameter is used to request that the SPLC be provided on the output tape records along with the city name. This option is requested via parameter and not via the control card.

The parameter appears on the execution JCL EXEC card. A sample appears below:

```
//MI200 EXEC PGM=MI200,PARM='SPLC=Y'
```

The valid parameter values are 'SPLC=Y' and 'SPLC=N'. If the parameter is omitted 'SPLC=N' will be assumed. 'SPLC=Y' causes the SPLC to appear on the output tape records.

Record Description: Input Control Record

A control record must be provided to control processing. The control record may be the first record on the input file, or it can be provided on a separate control card override file. If the control card is present in both places, the one from the override file is used.

The record length is 80 for format one, and 116 for format two, the same as the origin-destination records. If the control card file is used, the record length is 80. The fields of the record are defined as follows:

Position	Length	Name	Description
1	1	Control ID	Must be 'K'. Identifies record ID as a control record.
2	1	Input/output identifier	Should be 'I' for input.
3	2	Application identifier	Must be 'BM' for BATCH/MILEAGE or 'BZ' for BATCH/ZIP CODE inquiries.
5	2	Input format	'01' for format one, or '02' for format two.
7	1	Guide indicator	Valid values are 'A' and 'B'. Use 'A' to request previous guide. Use 'B' to request current guide.
8	50	Client name	Mandatory
58	4	Client ID	Valid values space or client identification code to be placed on MI520D billing records.
62	*	FillerSpaces.	

* This field is 19 bytes long for format one and 55 bytes long for format two.

Record Description: Format One Input Record

The definition of the format one input record is as follows. Note that the origin/destination indicator may be followed by an SPLC, a state/city/county or a ZIP Code.

Position	Length	Name	Description
1	1	Origin/Destination	Must be 'O' for origin or 'D' for destination indicator

Position	Length	Name	Description
2	3	ZIP Code	Three digit ZIP Code. Leading and trailing zeros must be supplied. Not used if SPLC or state/city is used.
2	9	SPLC	Must be nine digits. Leading and trailing zeros must be present. Not used if state/city is present.
2	2	State	Two character state/province code.
2	5	5-Digit ZIP Code	5-Digit ZIP Code
4	40	City	City name or city with county qualifier preceded by a comma. See note below regarding county qualifiers.
44	16	County	Full county name or two character county qualifier. County is not needed if SPLC or ZIP Code is used.
60	21	User	This data will appear on the Comments tape data record.
Total Length:	**80**		

-

Note: If the county qualifier is entered in the city field, the following rules apply. The county qualifier does not have to immediately follow the city name. However, the comma followed by the county qualifier must not begin after position 16 in the city name. That means that the city name must be no longer than fifteen (15) characters. Only the standard abbreviations used in MileMaker processing may be used.

Examples of city/county qualifier combinations follow:

```
          1    1       2    2
1234512345012345012345012345            Comment
FOX ,PL                             Acceptable format.
FOX         ,PL                     Acceptable format.
FOX                 ,PL             Incorrect, comma is after
                                    position 16 in city name.
```

Note that the county qualifier may also be placed in the county field, or the full county name may be placed in the county field. If the county qualifier is present in the city field, the county field will be ignored, but it will appear on the output tape.

Record Description: Format Two Input Record

The definition of the format two input record is as follows. Note that the origin and destination may be either an SPLC, a state/city or a ZIP Code.

Position	Length	Name	Description
1	3	Origin ZIP Code Three digit ZIP Code.	Leading and trailing zeros must be present. Not used if origin state/city or SPLC is used.
1	9	Origin SPLC	Must be nine digits. Leading and trailing zeros must be present. Not used if origin state and city are used.
1	5	Origin 5-Digit ZIP Code	Five digit ZIP Code must be five digits.
1	2	Origin state	Two character state or province code.
3	40	Origin city	City name or city with a county qualifier preceded by a comma. See note regarding county qualifiers.
43	16	Origin county	Full county name or two character county qualifier.
59	3	Destination ZIP Code	Three digit ZIP Code. Leading and trailing zeros must be present. Not used if destination SPLC or state/city is used.
59	9	Destination SPLC	Must be nine digits. Leading and trailing zeros must be present. Not used if state and city are used for the destination.
59	5	Destination 5-Digit ZIP	5-Digit ZIP Code must be five digits. code
59	2	Destinationstate	Two character

Position	Length	Name	Description
			state/province code.
61	40	Destination city	City name or city with county qualifier preceded by a comma. See note regarding county qualifiers.
101	16	Destination county	Full county name or two character county qualifier.
Total Length:	**116**		

Record Description: Tape Output Control Record

A control record will precede the mileage output records. The record length is 139. The fields of the record are defined as follows:

Position	Length	Name	Description
1	1	Control record ID	Will be 'K'. Identifies the record as a control record.
2	1	Input/output identifier	'O' for output.
3	2	Application identifier	'BM' for BATCH/MILEAGE or 'BZ' for BATCH/ZIP CODE.
5	2	Input format	'01' for format one, or '02' for format two.
7	1	Guide indicator	Valid values are 'A' and 'B'. Use 'A' to request previous guide. Use 'B' to request current guide.
8	50	Client name	As it appears on the input control record.
58	4	Client ID	As it appears on the input control record.
62	78	Filler	Contains blanks.

Record Description: Tape Output Copyright Records

Records containing copyright information will appear on the output tape preceding and following the data. To facilitate user handling of these records, a slash ('/') will

be placed in column one of each record. Each record is 139 bytes long. A copy of the text as it will appear on the tape follows.

```
/REPRODUCTION OR COPYING OF THIS
/MATERIAL IN ANY FORM OR MEDIUM IS
/PROHIBITED. USE OF THIS TAPE AND
/THE DATA IT CONTAINS IS UNDER THE
/AUTHORITY OF A LICENSE FROM RAND
/MCNALLY & COMPANY DISCLOSURE TO OR
/USE OF THIS TAPE OR DATA BY OTHER
/THAN LICENSEE, WITHOUT PERMISSION
/OF RAND MCNALLY & COMPANY, VIOLATES
/THAT LICENSE AND IS PROHIBITED.
```

Record Description: Tape Output Record (Format One)

This record is written to the output tape file by the batch mileage inquiry program if format one input is used.

Position	Length	Name	Description
1	1	Origin/Destination Code	'O' for origin or 'D' for destination.
2	2	State Code	Two character state or province code.
4	40	City	City name as it appears on the input, if present, otherwise, as it appears on the Geographic File.
44	16	County	If provided on the input.
60	3	ZIP Code	For ZIP Code inquiry the ZIP Code is written instead of the SPLC. The next six bytes will contain spaces.
60	9	SPLC	If provided on input or requested on the parameter.
60	5	5-Digit ZIP Code	If provided on the input.
69	4	Mileage	Four digit display numeric mileage appears on destination records. On origin records this field contains zeros.

Position	Length	Name	Description
73	1	Status Code	Appears on destination records only. C - Calculated Mileage S - Stored Mileage E -Error in Mileage Routine (bad Long/Lat, etc.). X -Edit Error (e.g. point not on Geographic file). See Error Listing for error description.
74	21	Comments	User comments from input record.
95	45	Filler	Contains blanks
Total Length:	**139**		

Record Description: Tape Output Record (Format Two)

This record is written to the output tape file by the batch mileage inquiry program if format two input is used.

Position	Length	Name	Description
1	2	Origin State	Two character state/province code.
3	40	Origin City	City name as it appears on the input, if present, otherwise as it appears on Geographic File.
43	16	Origin County	If provided on the input.
59	3	Origin ZIP	Code For a ZIP Code inquiry the ZIP Code is written instead of the SPLC. The next six bytes will contain spaces.
59	9	Origin SPLC	If provided on input or if requested on the parameter.
59	5	Origin 5-Digit ZIP Code	If provided on the input.
68	2	Destination State	Two character state or province code.
70	40	Destination City	City name as it appears on the input, if present, otherwise, as it appears on Geographic file.
110	16	Destination County	If provided on the input.
126	3	Destination ZIP Code	For a ZIP Code inquiry, the ZIP Code is written instead of the SPLC. The next six bytes will contain spaces.
126	9	Destination SPLC	If provided on input or requested on the parameter.
126	5	Destination 5-Digit ZIP	If provided on the input.

Position	Length	Name	Description
135	4	Mileage	Four digit display; numeric mileage appears on destination records. On origin records this field contains zeros.
139	1	Status Code	C-Calculated Mileage S-Stored Mileage E-Error in Mileage Routine X-Edit Error. See Error Listing.
Total Length:	**139**		

Report Output Description

The mileage report lists up to 120 mileages on each page in three columns. There is a page break for each new origin. The city name as it appears on the Geographic file is shown on the report, regardless of the input type used (SPLC, ZIP Code or City Name).

The meaning of the status codes listed on the report are as follows:

S -The mileage is a stored mileage

C -Some portion of the mileage has been calculated because either the origin or the destination, or both, is not on the highway network.

E -There was an error encountered in obtaining the mileage. Consult your Customer Support Representative.

***** An edit error occurred. Refer to the accompanying error listing for further information.

If an edit error occurs for an origin, it and its destinations will not appear on the mileage report. In all other cases, all input points will appear.

The error listing shows all edit errors encountered. The messages are described in the Error Messages section of this document. **If an origin is in error, all of the associated destinations will appear on the error listing, even if some or all of the destinations were not in error.** If a destination is in error and the origin is not, only the destination(s) in error, along with the origin, will be shown on the error listing.

Error Messages

A list of messages appearing on the edit report or displayed in the PROGRAM MESSAGES section follows:

Blank Keypoint Code

This message is a warning; it is produced if the point is a non-keypoint.

Cannot Open Geo File

There is an error with the Geographic File or with one or both alternate index files.

City Name Needs County Qualifier

The county is required for the point but was not provided.

City/state Not Found

The requested city/state was not found on the Geographic File.

Control Record Company Name Blank

The customer's company name must appear in columns 8 thru 57 on the control record.

Control Record Application Not BM Or BZ

Columns 3 and 4 on the control record must contain the letters 'BM' or 'BZ'. 'BM' stands for 'Batch Mileage'. 'BZ' stands for 'Batch ZIP Code'.

Control Record Format Not Consistent With Input

If columns 5 and 6 contain the number '01', format 1 input must be provided using the INPUT1 DD statement in the JCL (see the JCL supplied on the distribution tape). If columns 5 and 6 contain '02', format 2 input must be provided using the INPUT2 DD statement in the JCL. If the control record format does not match the JCL, this message will appear.

Control Record Error--guide Must Be A Or B

Column 7 of the control record must contain an 'A' to request previous guide mileages, or 'B' for current guide mileages.

Control Record Error-previous Guide Not Allowed

Previous guide mileages were requested but the parameter file specifies that previous guide is not available.

Control Record Format Not Valid

Column 5 and 6 must contain the number '01' or '02'.

Control Record Type Not K

The first column of the input control record must be 'K'.

County Qualifier Invalid

The county supplied is invalid for the requested city. This means that the city/county combination could not be found on the Geographic File.

Deactivated Point

Mileages to one or more of the points on the input record are no longer obtainable.

Geographic File Read Error

This error should be referred to your Data Processing systems group.

Input-output Code Not I

Column two on the control record must be 'I' (for input).

Invalid Origin

This message appears whenever an origin point is in error. The message will appear next to all of the destinations associated with the origin in error. If the destination is in error that error message will appear instead of this message.

Invalid SPLC

The requested SPLC could not be found on the Geographic File.

Invalid ZIP Code

The ZIP Code was not numeric or could not be found on the ZIP Code Master File.

Missing Destination

For single record input, the destination field is blank.

Missing Origin

1. For multiple record input, the first record is not an origin record.
2. For single record input, the origin is blank.

No Errors Detected

Run was completed successfully and no edit errors were found.

Origin/destination Code Invalid

For multiple record input only, column 1 must be an alpha 'O', for origin, or 'D', for destination.

Duplicate Cities for ZIP Code

The requested 5-digit ZIP Code has multiple cities use city/state or SPLC input.

MILEAGE/ZIP INTERFACE

Chapter 10

Chapter Contents

Mileage/ZIP Interface Requirements

The Mileage/ZIP interface is designed to interact with other application programs to return HHG mileages when given either city/state, SPLC or ZIP Code.

The following describes the interface areas to be used by programs interfacing with the mileage routines. The actual interface areas included are shown in Appendix A. This narrative discusses the values which should be passed into the interface areas and the values which will be returned by MileMaker.

Online

The user program must contain a 2966 byte area in the working storage section (MI-INTERFACE-COMMAREA). Refer to Appendix D. This area should be passed by using the COMMAREA and LENGTH options of the LINK to MI110.

Upon completion of the mileage request, control is returned from the mileage interface module MI110.

The online user program must be compiled with the following options:

```
CICS COBOL for OS/390 : RENT, NODYNAM, TRUNC (BIN), NOOPT,
```

Batch

The call to MI210 should be 'USING' the interface areas shown in Appendix D. The batch user program must be compiled with the following options:

```
BATCH COBOL for OS/390 : RENT, DYNAM, TRUNC (BIN), NOADV,
NOOPT
```

Field Definitions

The following are the detail data field definitions:

MI-REQUEST-TYPE. This field contains a code indicating the type of service requested from the MileMaker modules. This field should be initialized with an 'M' for Mileage inquiries and a 'Z' for ZIP inquiries. On reaching the end of the input file in batch mode, a final call to MI210 with this field set to 'C' should be made to permit orderly shutdown of the files. The values entered will be returned in the field.

MI-MILEAGE-IND. This field indicates whether the Current or Previous Guide mileage is required. This field should contain an 'A' if the Previous Guide is required , or a 'B' if the Current Guide is required. No ZIP mileages are available for Previous Guides. If this field is not filled, the Current Guide is used.

MI-ERROR-TYPE. This field is used to indicate whether a global error occured during the MileMaker processing. It is contrasted with the MI-ERROR-CODE field in each entry which indicates that an error has occurred which relates to that individual point entry. This field should be initialized to spaces. Upon return from MileMaker, this field should be checked first for an error indication prior to processing the returned interface area. When other than a space is returned, the inquiry was not processed. Return values are:

NF	At least one point was not found on the Geographic or User Conversion files.
NP	Previous Guide was requested but is not supported.
NE	The number of entries field was invalid -- either it was not numeric, less than 2 or greater than 40.
ST	At least one entry has an 'E' status value in the MI-STATUS field. (see MI-STATUS below.)
CE	A file error has occurred. In this case, the first entry in the table will contain the file in question in the MI-ERROR-FILE field.
Spaces	No error occurred

MI-NBR-ENTRIES. Number of used entries in the table. Since the new interface area is fixed length, there will always be forty (40) available entries. This field must be a valid numeric value between two (2) and forty (40), inclusive, or the NE error type will be set and the inquiry will not be processed. The unused entries will be returned unchanged should the user module require utilization of them.

MI-STATE-CITY. This group item should be initialized to spaces if SPLC is to be used for the mileage inquiry requested. In this case, the MI-SPLC field should be filled with a valid packed decimal value for the key to be used in the inquiry. In all cases, the filler at the back of this group item should be spaces.

MI-STATE. The standard two character state or province code. Should be blank when requesting SPLC or ZIP inquiries.

MI-CITY. This is an eighteen (18) character field containing the city name. If abbreviating is required due to excessive city name length, only the abbreviations in the User Manual are valid. This field should be blank when requesting an SPLC inquiry. See MI-CTY below for those cities requiring a county qualifier. For ZIP inquiries, the three digit numeric ZIP Code should be entered in this field.

MI-CY. This is a fifteen (15) character field containing the city name if the city occurs more than once in the state.

MI-CTY. For those city names occurring more than once in a given state, this field will contain a comma in position 1, followed by county code in positions 2 and 3. For all other cities, the city name may extend into this field.

MI-ZIP5. The 5-digit ZIP Code. This field must be a 5-digit number. If state, city, SPLC or 3-digit ZIP are being used this field should contain spaces.

MI-SPLC. The SPLC of the entry point. This field must be a valid packed decimal numeric value. If state and city or ZIP are being used for the inquiry, this field

should contain packed decimal zeros. If state and city are spaces and this field is either non-numeric or zeros, a 'NF' error type will be returned.

MI-ERROR-CODE. This field should be initialized to zeros. It will either be returned as such or will be returned with the following codes if an error in this particular entry is encountered:

01 City name needs county qualifier

02 City/state not found

03 File error has occurred (See MI-ERROR-FILE to determine which file is involved)

04 Deactivated geographic point

05 5-digit ZIP Code has multiple cities

MI-ERROR-FILE. This field should be initialized to spaces. 'GE010D' will be returned if a file error has occurred while accessing the entry on the Geographic file. 'UM010D' will be returned if a file error has occurred while accessing the User Conversion file. 'PM010D' will be returned if a file error has occured while accessing the PARAMETER file.

MI-STATUS. This field indicates either how the returned mileage was determined or that an error was encountered during the mileage processing. It should be initialized to spaces before linking to the interface module. If there was an edit error, such as the city/state not found on the Geographic File, an 'E' will be returned in this field. Check the MI-ERROR-CODE field for the error type. Valid return values are:

S Stored mileage

C Calculated mileage

E Mileage error encountered

MI-MILEAGE. This packed decimal field should be initialized to zeros. If an error is encountered, zeros will be returned. The point to point mileage will be returned in each origin point entry. The total trip mileage will be returned in the last used entry (final destination).

MI-AUD-INTERNAL-KEY. This pair of fields should be set to zeros before linking to the interface module. They are used internally in the MileMaker routines. If a non-keypoint entry is passed as the origin and/or destination of the trip, the internal key of the keypoint or node used for the origin and destination mileage determination will be returned in these fields, respectively. If the origin and/or destination of the trip is a keypoint or node, the corresponding field will be returned as zeros.

MI-AUD-MILES. This pair of fields should be set to zeros before linking to the interface module. They are used internally in the MileMaker routines. If a non-keypoint entry is passed as the origin and/or destination of the trip, the calculated portion of the total mileage will be returned in these fields. If the origin and/or

destination of the trip is a keypoint or node, the corresponding field will be returned as zeros.

Accounting Definition Field

AC-CLIENT-ID. This field should be initialized to spaces or low-value, if batch billing records are not to be created on the optional MI520D file. If billing records are to be identified, any field value other than space or low-value in this area will also be placed in the MI520D records.

Note: This field is not used in online programs. In online, the operator ID from sign-on is applied as the client ID on the optional MI520D records.

On-Line Access To MileMaker Via the User Menu

The On-line MileMaker programs are designed to allow entry to MIM001, MIM100, MIM103, and MIM310 screens from other application programs, and return (XCTL) to an application program after normal MileMaker processing is complete.

The user program must contain a 132 byte area in the Working Storage section (TWA-WORK-AREA). The name of the user program, a transfer control indicator, and an optional valid user ID must be moved into TWA-PROGRAM-NAME, TWA-XCTL-IND, and TWA-CLIENT-CODE before a transfer control (XCTL) to program MI001. The COMMAREA and LENGTH options are used to pass data to MI001. A sample command Level CICS program is provided in Appendix D.

By adding this code, the user has the option to select MileMaker inquiry functions Point Pair (MI100), Origin (MI103), and Route Inquiry (MI310) directly from their screen.

BATCH ROUTE INQUIRY

Chapter 11

Chapter Contents

General Description

The route inquiry program produces a printed report containing routes and /or state mileage breakdowns. The input points are edited and an error report is produced for points in error. For valid points, the route is obtained and printed.

The input data consists of an origin record followed by one or more destination records. Optionally, additional sets of origin/ destination records may be included. If multiple destinations are used for an origin, the last destination is the final destination. The other destinations are treated as stopoffs. There can be up to 39 destinations (38 stopoffs and the final destination) for one origin.

All city/states and SPLCs (origins and destinations) are checked to ensure that they are on the Geographic File. All points in error are listed on the Error Report. If an origin is in error, all of the associated destinations will appear on the report, If the origin or any destination is in error, no route will be produced for that set of points.

Program MI410 is provided on the SYSTEM DISTRIBUTION TAPE. It is contained in dataset "TDM.LDL.T". Sample JCL for the execution of MI410 is contained in dataset "TDM.JCL.T", also on the SYSTEM DISTRIBUTION TAPE. The member name is MI410X.

The programs which make up the batch route system follow:

Main Program:	MI410	
Subprograms:	GE101	MI225 GE102
	MI201	MI240
	MI215	MI420
	MI220	MI430

The MI410 load module on the Distribution tape was compiled under Enterprise COBOL and dynamically calls its subprograms and Enterprise COBOL system routines.

Control Record Description-Input Control Record

A control record must be provided to control processing. The control record may be the first record on the input file, or it can be provided on a separate control card override file. If the control card is present in both places, the one from the override file is used.

The record length is 80, the same as the origin-destination records. If the control card file is used, the record length is also 80. The fields of the record are defined as follows:

Position	Length	Name	Description
01	01	Control record ID	Must be 'K'. Identifies record as a control record.
02	01	Input/output Identifier	Should be 'I' for input.
03	02	Application Identifier	Must be 'BR' for BATCH/ROUTE inquiries.
05	01	Guide Indicator	Valid values are 'A' and 'B'. Use 'A' to request previous guide. Use 'B' to request current guide.
06	01	Function Code	This code specifies the request type A-Practical route only B-Practical state mileage breakdown (SMB)-Practical route and SMB D-Shortest distance route E-Shortest distance SMB F-Shortest distance route and SMB G-Shortest distance full route and SMB
07	50	Client name	Mandatory.
57	24	Filler	Spaces

Input Record Description

The definition of the input record is as follows. Note that the origin/destination indicator may be followed by an SPLC or a state/city/county.

Position	Length	Name	Description
01	01	Origin/Destination indicator	Must be 'O' for origin or 'D' for destination
02	09	SPLC	Must be nine digits. Leading and trailing zeros must be present. Not used if state/ city is present.
02	05	5-digit ZIP Code	Must be five digits. Not

Position	Length	Name	Description
			used if state/city is present.
02	02	State	Two character state/province code.
04	40	City	City name or city with county qualifier preceded by a comma. See note below for county qualifiers.
44	16	County	Full county name or two-character county qualifier. County is not needed if SPLC is used.
60	21	Filler	Field may be used for user comments. It is not edited or used.
Total Length:	**80**		

-

Note:

If the county qualifier is entered in the city field, the following rules apply. The county qualifier does not have to immediately follow the city name. However, the comma followed by the county qualifier must not begin after position 16 in the city name. That means that the city name must be no longer than fifteen (15) characters. Only the standard abbreviations used in MileMaker processing may be used.

```
          1    1      2    2
12345123450123450123450123 45          Comment
FOX ,PL                              Acceptable format.
FOX          ,PL                     Acceptable format.
FOX                ,PL               Incorrect, comma is after
                                     position 16 in city name.
```

Note that the county qualifier may also be placed in the county field, or the full county name may be placed in the county field. If the county qualifier is present in the city field, the county field will be ignored, but it will appear on the output tape.

Error Messages

A list of messages appearing on the edit report or displayed in the PROGRAM MESSAGES section follows.

City Name Needs County Qualifier

The county is required for the point but was not provided.

City/state Not Found

The requested city/state was not found on the Geographic File.

Control Record Company Name Blank

The customer's company name must appear in columns 7 thru 56 on the control record.

Control Record Application Not BR

Columns 3 and 4 on the control record must contain the letters 'BR'. 'BR' stands for 'Batch Route'.

Control Record Error-guide Must Be A Or B

Column 5 of the control record must contain an 'A' to request previous guide routes, or 'B' for current guide routes.

Control Record Error-previous Guide Not Allowed

Previous guide routes were requested but the Parameter File specifies that previous guide is not available.

Control Record Type Not K

The first column on the control record must be 'K'.

Deactivated Point

Mileages to one or more of the points on the input record are no longer obtainable. The points are kept on thbe Geographic file for historical auditing purposes only.

Ge101 Open Status = 99

There is an error with the Geographic File or with one or both alternate index files. 99 is the VSAM file status code. Notify your MileMaker technician.

Input-output Code Not I

Column two on the control record must be 'I' (for input).

Invalid City

The city on the input record is blank.

Invalid Function

Either the function is an invalid character or the function is not available. If the function is not available, the next message will further explain the error. Valid function codes are 'A','B','C','D','E','F' or 'G'.

Invalid Origin

This message appears whenever an origin point is in error. The message will appear next to all of the destinations associated with the origin in error. If the destination is in error that error message will appear instead of this message.

Invalid SPLC

The requested SPLC could not be found on the Geographic File.

Invalid State

The state code on the input record is blank or not alphabetic.

More Than 39 Destinations

Only 39 destinations are allowed for each origin. The origin listed on the error report exceeded that limit. Only the first 39 destinations are listed on the report.

No Errors Detected

Run was completed successfully and no edit errors were found.

Origin/destination Code Invalid

Column 1 must be an alpha 'O', for origin, or 'D', for destination.

Previous Guide Not Available On Parm File

Previous guide was requested on the control card but is not available on the Parameter File MI115 record.

Previous Guide Practical Not Available-using Current Guide

Previous guide was requested on a practical route. Only current Guide practical is available. Processing will continue using current Guide.

Require At Least Two Points

An origin or destination is missing from the input.

Duplicate Cities for 5-digit ZIP Code

The requested 5-digit ZIP Code has multiple cities. Use city/state or SPLC as imput.

PRACTICAL AND SHORTEST ROUTING INTERFACE

Chapter 12

Chapter Contents

Interface Requirements

The Practical and Shortest Distance Routing interface is designed to interact with other application programs to return the **State Mileage Breakdown for any given Practical or Shortest Distance Route.**

There are two interface areas used in processing the state mileage request. The User Interface Area is used to specify the points in the route. The State Mileage Table, upon return to the calling program contains the state mileage breakdown for the points in the route. The following describes the interface areas to be used by programs interfacing with the routing routines. The actual interface areas included are shown in Appendices E. The narrative discusses the values which should be passed into the interface area and the values which will be returned by MileMaker.

Online

The user program must contain a 4992 byte area in the working storage section (MI-INTERFACE-COMMAREA). This area should be passed by using the COMMAREA and LENGTH options of the LINK to MI335. The MI-INTERFACE-COMMAREA contains two parts: the first part contains a 2332 byte User Interface area; the second part contains a 2660 byte state mileage table. (Refer to Appendices E.)

Upon completion of the route request, control is returned from the route interface module MI335.

The online user program must be compiled with the following options:

```
CICS COBOL for OS/390: RENT, NODYNAM, TRUNC (BIN), NOOPT
```

Batch

The call to MI435 uses two areas: the User Interface Area and the state mileage table. The following compile options must be used:

```
BATCH COBOL for OS/390: RENT, DYNAM, TRUNC(BIN), NOADV, NOOPT,
```

Data Field Definitions

The following are the detail data field definitions for the User Interface area:

MI-REQUEST-TYPE. This field contains a code indicating the type of service requested from the MileMaker modules. This field should be initialized with a 'P' for Practical Route State Mileages and an 'S' for Shortest Distance Routing State Mileages. On reaching the end of the input file in batch mode, a final call to MI435 with this field set to 'C' should be made to permit orderly shutdown of the files. The values entered will be returned in the field.

MI-MILEAGE-IND. This field indicates whether the Current or Previous Guide mileage is required. It should contain an 'A' if Previous Guide is required , or 'B' if Current Guide is required. If field not filled in, Current Guide is used.

MI-ERROR-TYPE. This field is used to indicate whether a global error occurs during the MileMaker processing. It is contrasted with the MI-ERROR-CODE field in each entry which indicates that an error has occurred which relates to that individual point entry. This field should be initialized to spaces. Upon return from MileMaker, this field should be checked first for an error indication prior to processing the returned interface areas. When other than spaces are returned, the inquiry was not processed. The return values are:

Spaces	No error occurred
NF	At least one point was not found on the Geographic or User Conversion files.
NP	Previous Guide was requested but is not supported.
NS	Shortest Route was requested but is not supported.
NT	Practical Route requested but is not supported.
NE	The number of entries field was invalid -- either it was not numeric, less than 2 or greater than 40.
CE	A file error has occurred. In this case,the first entry in the table will contain the file in question in the MI-ERROR-FILE field.
ST	At least one entry encountered a mileage error; for a description, refer to MI-RTE-ERROR-MSG.

MI-NBR-ENTRIES. Number of used entries in the table. Since the interface area is fixed length, there will always be forty (40) available entries. This field must be a valid numeric value between two (2) and forty (40), inclusive, or the NE error type will be set and the inquiry will not be processed. The unused entries will be returned unchanged should the user module require utilization of them.

MI-STATE-CITY. This group item should be initialized to spaces if SPLC is to be used for the mileage inquiry requested. In this case, the MI-SPLC field should be filled with a valid packed decimal value for the key to be used in the inquiry. In all cases, the filler at the back of this group item should be spaces.

MI-STATE. The standard two character state or province code. This field should be blank when requesting an SPLC inquiry.

MI-CITY. This is an eighteen (18) character field containing the city name. If abbreviating is required due to excessive city name length, only the abbreviations in the User Manual are valid. This field should be blank when requesting an SPLC inquiry. See MI-CTY below for those cities requiring a county qualifier. For ZIP inquiries, the three digit numeric ZIP Code should be entered in this field.

MI-CTY. For those city names occurring more than once in a given state, this field will contain a comma in position 1, followed by county code in positions 2 and 3. For all other cities, the city name may extend into this field.

MI-SPLC. The SPLC of the entry point. This field must be a valid packed decimal numeric value. If state and city or ZIP are being used for the inquiry, this field should contain packed decimal zeros. If state and city are spaces and this field is either non-numeric or zeros, a 'NF' error type will be returned.

MI-ZIP5. The 5-digit ZIP Code. This field must be a 5-digit number. If city/state or SPLC is used this field should be spaces.

MI-ERROR-CODE. This field should be initialized to zeros. It will either be returned as such or will be returned with the following codes if an error in this particular entry is encountered.

01	City name needs county qualifier
02	City/state not found
03	File error has occurred (See MI-ERROR-FILE to determine which file is involved)
04	Deactivated geographic point
05	Duplicate cities for 5-digit ZIP Code

MI-ERROR-FILE. This field should be initialized to spaces. A 'GE010D' will be returned if a file error has occurred while accessing the entry on the Geographic file. A 'UM010D' will be returned if a file error has occurred while accessing the User Conversion file. A 'PM010D' will be returned if a file error has occurred while accessing the PARAMETER file.

MI-CALLING-PROGRAM. Initialize with name of user calling program.

MI-TERM-ID. Online use only; Terminal identification of the user terminal.

MI-RTE-STATUS. Returned from route display. Value 'E' if route cannot be completed. Initialize as space.

MI-RTE-ERROR-MSG. If MI-RTE-STATUS='E', field contains the associated error message for the error.

MI-RTE-TOTAL-MILEAGE. Total mileage for the route.

MI-RTE-TOLL-MILEAGE. Total toll mileage for the route.

MI-TS-QUE-NAME. Online use only; the storage que name for the route. Should be initialized to spaces.

MI-TS-NUM-PAGES. Online use only; the number of pages in the route display.

MI-ST-MLG-TABLE-ADDR. Should be initialized to zeros.

State Mileage Table

The State Mileage Table for batch is defined in the Calling Program's Working Storage Section. It is passed to the Route Interface Module (MI435) via the Second

Parameter in the Using clause on the call to MI435. Refer to the Appendix E for a working sample Batch program.

The State Mileage Table for the online interface is defined in the Working Storage Section. Refer to Appendix E for a working sample online program.

Upon return to the Calling Program the State Mileage Table will contain the state mileage breakdown for the points in the route. The table contains an entry for each state defined to the MileMaker System. The state code is implied by the relative position within the table. The relative position is an index to the State Name Table (Appendix E). The State Name Table contains the full state name and the accepted abbreviation for each state. The following is a description of the data returned from the routing routines. Refer to Appendix E for the actual state mileage table record layout.

ST-RTE-TYPE. Returned - 'P' for Practical Route. 'S' for Shortest Route.

ST-COUNT. Returned - The number of states traveled in the route. (Includes both ST-MI and ST-MI-BOBTAIL fields).

ST-MI. Returned - Total mileage for the state; zero if not used.

ST-MI-LOADED. Currently not being used.

ST-MI-EMPTY. Currently not being used for all except the first entry in the table. For Practical Routing only, the first table entry contains the total trip time in hours and minutes, formatted as HHHMM.

ST-MI-TOLL. Returned - Total toll miles for that state. Non-toll miles can be derived by subtracting ST-MI-TOLL from ST-MI.

ST-MI-BOBTAIL. Returned - Value of +99999 if total mileage traveled in a state is less than 1 mile; ST-MI field will contain zeros. Zero if state was not traveled.

Interface Requirements with Detail Route

The Practical and Shortest Distance Route with Detail Routes is designed to interact with other application programs and return state mileage breakdown and detailed route information for any Practical or Shortest Distance Route inquiry.

There are three interface areas used in processing the state mileage and the detail route request; the user interface area is used to specify the locations in the route; the state mileage table and the route detail table, upon return to the calling program, contain the state mileage breakdown and the detailed route information for the locations in the route. The interface areas to be used by programs interfacing with the detailed route routines are described above. The actual interface areas are shown in Appendix E).

Online

The user program must contain a 4992 byte (MI-INTERFACE-COMMAREA) and a 1399 byte (Route-DISPLAY-AREA) in the working storage section. The MI-INTERFACE-COMMAREA should be passed by using the COMMAREA and LENGTH options of the LINK to MI7335. (Refer to Section 1 "Online Requirements".)

Upon completion of the route request, control is returned from the route interface module MI7335. The detailed route will be contained in a Temporary Storage Queue

("MD00xxxx" or "MT00xxxx" where xxxx = EIBTRMID).

It is the user's responsibility to delete this queue upon completion of processing. (Refer to Section 1 for online compile options.)

Batch

The call to MI7435 uses three areas: the User Interface Area, the State Mileage Table, and the Route-Display Area. (Refer to Section 1 for batch compile options.)

COMPONENT FILE DELETION PROGRAM

Chapter 13

Chapter Contents

Description of MI9390D Program

The Component File Delete Program offers a method of performing mass deletions from the Component File (MI390D). This program will normally be run only at the direction of Rand McNally. It is used to delete all MI390D records containing mileages to or from a specified point.

Any time a change is made which may affect the mileage to or from a point, this program should be run. The alternative is to completely delete the MI390D file and start with a fresh file. Possible changes include:

- A change to the corporate limit field on the Geographic File.
- A change to the longitude or latitude. This may even affect some mileages to a keypoint if the keypoint longitude/latitude is changed.
- A change to the vicinity code on the Geographic File.
- A change to a MI605M record for a Secondary Keypoint.

This program may also be used to delete records from the MI391D file, the Practical Routing Component File.

Sample JCL For MI9390D

Sample JCL to run the Component file delete follows. See member MI9390DX on the distribution JCL dataset for the actual JCL.

```
//MI9390D JOB
//*
//*************************************************************
//* DELETE RECORDS FROM THE BASIC MILEAGE COMPONENT FILE,
//* MI390D OR FROM THE PRACTICAL ROUTING COMPONENT FILE,
//* MI391D
//*************************************************************
//*
//STEP1    EXEC PGM=IDCAMS
//SYSPRIN  DD SYSOUT=A
//MI390D   DD DSN=TDBV.PMI390D.P, --BASIC OR PRAC. COMP FILE
//            DISP=SHR
//SYSIN    DD *
 VERIFY FILE (MI390D)
/*
//STEP2 EXEC PGM=MI9390D
```

```
//MI390D   DD DSN=TDBV.PMI390D.P, --BASIC OR PRAC. COMP FILE
//            DISP=SHR
//GE010D   DD DSN=TDBV.PGE010D.P, --GEO BASE (STATE,CITY)
//            DISP=SHR
//GEO10D1  DD DSN=TDBV.PGE010D.PATH1, --GEO ALT1 (SPLC)
//            DISP=SHR
//GEO10D2  DD DSN=TDBV.PGE010D.PATH2, --GEOALT2 (INTERNAL)
//            DISP=SHR
//STEPLIB  DD DSN=USER.LINKLIB, --USER LINKLIB
//            DISP=SHR
//SYSPRINT DD SYSOUT=A
//SYSOUT   DD SYSOUT=A
//SPLCIN   DD *
//* PLACE INPUT SPLC CARDS HERE.
//* ONLY ONE SPLC PER CARD
//* SPLC STARTS IN COL ONE AND ENDS IN COL NINE.
//* A MAX OF 5 SPLC CARDS CAN BE USED.
/*
//
```

Input SPLC File Record Format

The records on the SPLC File contain the SPLC's of those points to be deleted from either the Basic Component File, MI390D, or the Practical Component File, MI391D.

The input SPLC File contains a maximum of five records. Each record can have only one SPLC.

The record format is as follows:

Position	Length	Name	Description
01	09	SPLC	Must be 9 digits. Leading and trailing zeros must be included.
10	80	Not Used	May contain User comments such as city, state.
Total record length	**80**		

The SPLC must exist on the Geographic File, Otherwise it is considered as an error and not processed. The SPLC to be deleted is the SPLC, not the internal key.

MI9390D calls the external modules GE101 and MI201 in order to verify the existence of the SPLC's on the Geographic File.

List Of Counts Produced by MI9390D Program

The following counts are printed after completion of the job:

```
NUMBER OF RECORDS READ FROM MI390D FILE
NUMBER OF RECORDS DELETED FOR SPLC 1
NUMBER OF RECORDS DELETED FOR SPLC 2
NUMBER OF RECORDS DELETED FOR SPLC 3
NUMBER OF RECORDS DELETED FOR SPLC 4
NUMBER OF RECORDS DELETED FOR SPLC 5
TOTAL NUMBER OF RECORDS DELETED
```

The reference to the MI390D file in the first message above refers to the DDNAME/DLBL in the JCL. The count will give the number of records read from the input file whether the MI390D or MI391D file is used.

DATA FILE DESCRIPTIONS

Chapter 14

Chapter Contents

Introduction

The following pages contain descriptions of MileMaker data files. Included are descriptions of all required and optional files for the Basic System, Shortest Distance Routing System and Practical Routing System.

Basic System	**Current Guide**	**Previous Guide**
Geographic File	GE010DP	GE011DP
Geographic Update Log File	GE020D	
3-Digit ZIP Code Mileage File	MI015D	
Locator File	MI020D	MI022D
Mileage/Route File	MI140B	MI140A
Traversal File	MI145B	MI145A
3-Digit ZIP Code Master File	MI105D	
Surrounding Blocker File	MI305D	MI307D
Short Distance Mileage File	MI340M	MI342M
Cross Reference File	MI350D	MI352D
Component File	MI390D	
Secondary Key Point Mileage File	MI605M	MI607M
User Conversion File	UM010D	
Billable Point Pairs File	MI520D	
Parameter File	PM010D	
5-Digit ZIP Reference File	MI882D	MI883D
Shortest Distance Routing System		
Highway Blocker File	MI320D	MI322D
Secondary Key Point Route File	MI605R	MI607R
Short Distance Route File	MI340R	MI342R
Practical Routing System		
Locator File	MI021D	
Time/Route File	MI141B	
Surrounding Blocker File	MI306D	
Highway Blocker File	MI321D	
Practical Distance Route file	MI341R	
Cross Reference File	MI351D	

Component File	MI391D
Secondary Key Point Route File	MI605R

Note: a single copy of the MI605R file is used for both Shortest and Practical Routing.

Geographic File (Guide 19 Basic)

File Number:	GE010DP (Primary Cluster)
	GE010D1(Alternate Key 1 File)
	GE010D2 (Alternate Key 2 File)
Data Base Name:	TDBV.PGE010D.P
	TDBV.PGE010D.PATH1
	TDBV.PGE010D.PATH2
Number Of Records:	154,282
File Size:	24 MB
Record Key:	Primary - State, City
	Alternate 1 - SPLC
	Alternate 2 - INTERNAL KEY
Record Format:	Fixed
Record Length:	124

Record Contents

This file contains one record for every city or town in the U.S. and Canada. Each record contains city/state name, splc, latitude, and longitude. keypoint code, keypoint number (if applicable), corporate limit keypoint number and vicinity code for the current and previous guides are included.

If the point is in a keypoint area (the orange shaded or striped area on the mileage guide map) the corporate limit field contains the keypoint number of the keypoint area. All geo records for HHG keypoints (i.e. records which have keypoint code of 'k') have their keypoint number in the corporate limit field regardless of whether they have an associated keypoint area on the maps. Thus, all HHG keypoints are treated as if they have keypoint areas for rules processing.

The vicinity code is the map number of the vicinity map in the mileage guide on which the point appears, if any. If the point is in the area but does not appear on the map, it will not have a vicinity code.

Geographic File (Guide 18 Basic) for Previous Guide (G18)

File Number:	GE011DP (Primary Cluster)
	GE011D1(Alternate Key 1 File)
	GE011D2 (Alternate Key 2 File)
Data Base Name:	TDBV.PGE011D.P
	TDBV.PGE011D.PATH1
	TDBV.PGE011D.PATH2
Number Of Records:	Approximately 143,590
File Size:	24 MB
Record Key:	Primary - State, City
	Alternate 1 - SPLC
	Alternate 2 - INTERNAL KEY
Record Format:	Fixed
Record Length:	124

Record Contents

This file contains one record for every city or town in the U.S. and Canada. Each record contains city/state name, splc, latitude, and longitude. keypoint code, keypoint number (if applicable), corporate limit keypoint number and vicinity code for the current and previous guides are included.

If the point is in a keypoint area (the orange shaded or striped area on the mileage guide map) the corporate limit field contains the keypoint number of the keypoint area. All geo records for HHG keypoints (i.e. records which have keypoint code of 'k') have their keypoint number in the corporate limit field regardless of whether they have an associated keypoint area on the maps. Thus, all HHG keypoints are treated as if they have keypoint areas for rules processing.

The vicinity code is the map number of the vicinity map in the mileage guide on which the point appears, if any. If the point is in the area but does not appear on the map, it will not have a vicinity code.

Geographic Update Log File (Guide 19 Basic)

File Number:	GE020D
Data Base Name:	TDBV.PGE020D.P

Number Of Records: Depends On Number Of Geographic File Updates File Size: Varies

Record Key: None. Entry Sequenced File.

Record Format: Fixed

Record Length: 146

Record Contents

There is a record written to this file whenever an add, a delete or a key change is performed to the geographic file. in the case of an add, the data in the new record is kept on the GE020D file. in the case of a key change, the old and new keys are retained. for a delete, a copy of the deleted record is kept.

In addition to the above data, there are indicators in the record showing the update reason and type of key change, as well as the date and time of the change.

3-digit ZIP Code Mileage File (Guide 5 Basic)

File Number: MI015D

Data Base Name: TDBV.PMI015D.P

Number Of Records: 900

File Size: 3 MB

Record Key: None. Records Retrieved By Relative Record Number. (See Below For Details.)

Record Format: Fixed

Record Length: 2000

Record Contents

1000 2-byte binary mileages. when using this file, the relative record number and offset of the correct mileage entry are derived from origin and destination "place number". the place number is an arbitrary number assigned to a ZIP Code, similar to a keypoint number. it can be found on the MI105D file, the ZIP Code master.

Locator File (Guide 19 Basic)

File Number: MI020D

Data Base Name: TDBV.PMI020D.P

Number Of Records: 383

File Size: 1 MB

Record Key:	Highest Longitude And Latitude In The Record
Record Format:	Fixed
Record Length:	2036

Record Contents

Each record contains 145 entries which contain the longitude, latitude, internal key and keypoint number (for a keypoint) of a node or keypoint. There is an entry on the file for each node and keypoint in the system. the records are used by the calculation routine to locate the closest node or keypoint to a non-keypoint for developing a calculated distance. The entries are arranged in ascending longitude/latitude sequence.

Mileage/route File (Guide 19 Basic)

File Number:	MI140B
Data Base Name:	TDBV.MI140B.P
Number Of Records:	36,792
File Size:	76 MB
Record Key:	None This Is A Relative Record File. The Record Number Is Calculated Using The From And To Key Point Numbers.
Record Format:	Fixed
Record Length:	4088

Record Contents

This File Contains Mileages from each keypoint to every other keypoint. It also contains the keypoint strings used to develop the routes between keypoints. Each keypoint except the last (keypoint number 6132) has data contained on six records. Each keypoint except number 1 shares its records with another keypoint. Key Point numbers less than 3067 have their data in the beginning of the records and numbers greater than 3066 have their data in the end of the records. Key Point 2 shares with 6131, 3 shares with 6130 and so on. Thus, while the records are logically variable, the physical organization is fixed.

Traversal File (Guide 19 Basic)

File Number:	MI145B
Data Base Name:	TDBV.MI145B.P
Number Of Records:	9868

File Size: . 1 MB

Record Key: Low Key Point Number, High Key Point Number.

Record Format: Fixed

Record Length: 8

Record Contents

This file contains traversals between adjacent keypoint pairs on the road network. The term "adjacent" means that there are no keypoints on the route between the two keypoints. If the route goes thru a keypoint area (one of the blue shaded areas on the mileage guide) a record will appear on the traversal file for the keypoint pair. The record contains the first and last traversal keypoint number only. This file is used in conjunction with the mileage/route file to determine whether a rules violation has occurred on a route.

3-Digit ZIP Code Master File

File Number: MI105D

Data Base Name: TDBV.PMI105D.P

Number Of Records: 900

File Size: . 1 MB

Record Key: 3-Digit ZIP Code.

Record Format: Fixed

Record Length: 31

Record Contents

Each record contains the state and city names and the splc which correspond to the ZIP Code. there is also a "place number" on each record. the place numbers are assigned sequentially to each ZIP Code. they are used to calculate record numbers to retrieve mileages from the ZIP Code mileage file (MI015D).

Surrounding Blocker File (Guide 19 Basic)

File Number: MI305D

Data Base Name: TDBV.PMI305D.P

Number Of Records: 51,610

File Size: 5 MB

Record Key: Internal Key Of The Node To Which The Record Belongs

Record Format: Variable

Record Length: 16-2207

Record contents

There is one record for each node on the geographic file. The base portion of the record contains the record key and the number of blocker entries. The blockers must be keypoints. There is an 9 byte entry for each blocker. Each entry contains the blocker keypoint number, mileage from the node to the blocker, corporate limit keypoint numbers of keypoint areas traversed, if any, and a border crossing flag.

There may be one or two corporate limit keypoint numbers. These would be for the first and last traversals between the node and the blocker. The blocker is also considered a traversal if it is an HHG keypoint (keypoint code 'K').

Short Distance Mileage File (Guide 19 Basic)

File Number: MI340M

Data Base Name: TDBV.PMI340M.P

Number Of Records: 4,823,298

File Size: 100 MB

Record Key: Low Internal Key, High Internal Key

Record Format: Variable

Record Length: 18 (see Note Below On Variable Segments)

Record Contents

The distance between nodes which do not have an intervening keypoint, and up to two corporate limit keypoint numbers. This file is needed because if the mileage between such points were derived using the blockers, the mileage would be too high.

Note: the record format is variable for compatability with the MI340R file, but none of the variable segments are used on the MI340M file. Those segments contain the road segments between the nodes. Thus, all records have the same length. The MI340R file can be used in place of the MI340M file by package clients which have routing to conserve disk space.

Cross Reference File (Guide 19 Basic)

File Number: MI350D

Data Base Name: TDBV.PMI350D.P

Number Of Records:	6,132
File Size: .	5 MB
Record Key: None.	Relative Record File Using Key Point Number As Record Number.
Record Format:	Fixed
Record Length:	9

Record Contents

There is a record for each Rand McNally and HHG key- point. Each record contains the keypoint number, internal key of the keypoint and the keypoint code. These records are used to relate the keypoint number to the internal key.

Component File (Guide 19 Basic, Optional)

File Number:	MI390D
Data Base Name:	TDBV.PMI390D.P
Number Of Records:	Variable
File Size:	User Defined
Record Key:	Low Internal Key, Low Key Point Code, High Internal Key, High Key Point Code
Record Format:	Variable
Record Length:	16-43

Record Contents

Prestored mileages and route components. When an initial mileage or route request is made for a given pair of points, a component record is added to the file if space is available. Subsequent requests for that point pair will use the component file to obtain the mileage to save processing time and resources. MileMaker will continue processing normally when MI390D space is exhausted or if the file is not allocated.

There are three record types; their use depends on the keypoint codes of the from and to cities. If one or both of the keypoint codes is a blank, type one is used; if one or both is a 2, type two is used; otherwise, type three is used. Types one and two contain additional information required for calculated points and secondaries which is not contained on type three records. A component record is not written for mileages between two HHG keypoints because the mileage is obtained by doing one read of the mileage file. The same is true for mileages between two Rand McNally keypoints or a Rand McNally keypoint and an HHG keypoint if there is no traversal between them.

Secondary Key Point Mileage File (Guide 19 Basic)

File Number:	MI605M
Data Base Name:	TDBV.PMI605M.P
Number Of Records:	Variable
File Size:	User Defined
Record Key:	Internal Key From Geographic File
Record Format:	Variable
Record Length:	33-943

Record Contents

Records on this file may be added by package clients to their file or requested by service clients. File size depends on the number of secondaries desired.

Each record has segments defining the surrounding points for the secondary keypoint. Each segment contains the internal key, keypoint number, if any, longitude and latitude of the surrounding point and the mileage from the secondary to the surrounding point. The first and last traversal keypoint numbers, if any, are also included.

User Conversion File (Guide 19 Basic)

File Number:	UM010D
Data Base Name:	TDBV.PUM010D.P
Number Of Records:	Variable
File Size:	User Defined
Record Key:	User ID, Type And User Code
Record Format:	Variable
Record Length:	40 - 261

Record Contents

The user-defined code and the internal key of the point that it redefines. This file can be used by a user to equate points on the Geographic file to customized names such as plant or office names. It is also used to store old names of Geographic records after a name change so that the point may be referenced by either the old or new name. The same is done for SPLC changes.

Rand McNally and their package clients use only the first 40 bytes of the User Conversion record.

Billable Point Pairs File (Guide 19 Basic, Optional)

File Number:	MI520D
Data Base Name:	TDBV.PMI520D.P
Number Of Records:	Varies
Record Key:	Client, SPLC, Month, Day And Transaction Type
Record Format:	Fixed
Record Length:	60

Record Contents

All billable point pairs used for billing clients. These include mileages, origins, Browses, Secondaries, User Conversion, Practical and Shortest inquiries and adds.

Parameter File (Guide 19 Basic)

File Number:	PM010D
Data Base Name:	TDBV.PM010D.P
Number Of Records:	Minimum Of 100, Plus User Added Records
File Size:	Client Dependent
Record Key:	Record Type, Rest Of Key Depending On Record Type (See Record Type Descriptions Below)
Record Format:	Fixed
Record Length:	80

Record Contents

System control parameters including valid states and provinces, mileage file options, passwords and user setup parameters. a detail description of each record type follows.

GE010 - State Definitions

Key:	State or province abbreviation
Contents:	The record contains the two-digit SPLC prefixes, the full name and longitude/latitude limits for each state and province. The long/lat limits define the farthest point in the state or province in each compass direction. These limits are used to edit new points for valid long/lats.

FR002 - Passwords

Key:	Password (up to 4 characters)

Contents: Security and file access codes for the passwords. Passwords are added as required.

MI115 - Milemaker Definitions

Key: Rest of key after record ID is blank. There is only one MI115 record on the Parm file.

Contents: Parameters controlling processing and defining the MileMaker environment.

FR011 - Default Client Attributes

Key: Terminal ID

Contents: At present merely contains the default client ID for the terminal. This ID will be placed in the user ID field during an inquiry if the operator does not enter a user ID. If there is no record for the terminal, the operator must enter a user ID.

FR012 - User Profile

Key: User ID

Contents: Route options for each user. These records are added as required. Trip accounting options are not used.

5-Digit ZIP Reference File

File Number: MI882D

Data Base Name: TDBV.PMI882D.P

Number of Records: 150,332

File Size: 4 MB

Record Key: 5-Digit ZIP Code, Sequence Number

Record Format: Fixed

Record Length: 60

Record Contents

The file contains those ZIP Codes recognized by the U.S. Post Office (approximately 43,000). Each record has a MileMaker geographic point assigned. Since ZIP Codes are not unique to a single point sequence numbers have been added to the record key in order to assign multiple geographic points to the same ZIP Code (adding approximately 34,000 records). For those ZIP Codes with unique points assigned, ZIP Code selection produces normal MileMaker processing as if the point were selected by city/state or SPLC. For the ZIP Codes with multiple points assigned the user must select by state/city or SPLC for normal MileMaker processing to occur.

5-Digit ZIP Reference File, for Previous Guide

File Number:	MI883D
Data Base Name:	TDBV.PMI883D.P
Number of Records:	161,494
File Size:	4 MB
Record Key:	5-Digit ZIP Code, Sequence Number
Record Format:	Fixed
Record Length:	60

Record Contents

The file contains those ZIP Codes recognized by the U.S. Post Office (approximately 43,000). Each record has a MileMaker geographic point assigned. Since ZIP Codes are not unique to a single point sequence numbers have been added to the record key in order to assign multiple geographic points to the same ZIP Code (adding approximately 34,000 records). For those ZIP Codes with unique points assigned, ZIP Code selection produces normal MileMaker processing as if the point were selected by city/state or SPLC. For the ZIP Codes with multiple points assigned the user must select by state/city or SPLC for normal MileMaker processing to occur.

Highway Blocker File (Shortest)

File Number:	MI320D
Data Base Name:	TDBV.PMI320D.P
Number Of Records:	1,394,619
File Size:	181 MB
Record Key:	Origin Internal Key, Destination Key Point Number
Record Format:	Variable
Record Length:	26-1088

Record Contents

The actual road segments which make up the route from the origin to the destination. The data on the segments includes the highway name, direction of travel, mileage, ending junction information and flags for toll roads or bridges. There are also segments containing long names for highways (such as Kennedy expwy for I90) or alias names.

There is an MI320D record containing the route from each node to each of the blockers for the node. This is why the internal key is used in the key. nodes have no keypoint numbers. The rest of the records contain keypoint to keypoint routes. There is not a record for each keypoint pair in the system. The route from keypoint A to keypoint B may go through other keyoints. in that case the MI140B file is used to obtain those keypoints. The MI320D records are then obtained using that list.

Secondary Key Point Route File (Shortest And Practical)

File Number:	MI605R
Data Base Name:	TDBV.PMI605R.P
Number Of Records:	Variable
File Size:	User Defined
Record Key:	econdary Key Point Internal Key Point, Surrounding Point, Internal Key.
Record Format:	Variable
Record Length:	44-204

Record Contents

This file contains the detailed information for constructintg routes from a secondary to its surrounding points. Each record contains up to six road segments detailing the route from the secondary to one of the surrounding points. The data for each segment includes highway name, direction of travel, and miles traveled on the segment.

Short Distance Route (AKB) File

File Number:	MI340R
Data Base Name:	TDBV.PMI340R.P
Number Of Records:	4,823,298
File Size:	1000 MB
Record Key:	Low Internal Key, High Internal Key
Record Format:	Variable
Record Length:	34-2318

Record Contents

Route description between nodes which do not have an intervening keypoint. The base portion of the records are the same as the MI340M file. If a client has routing, he may point the MI340M DD/DLBL statement in the online at the MI340R file. The data on the variable segments includes the highway name, direction of travel, mileage, ending junction information, and flags for toll roads or bridges. There are also segments containing long names for highways (such as Kennedy Expwy for I94) or alias names.

For nodes without intervening keypoints, the MI340R file is used in the same manner as the MI320D file. The variable segments supply all of the information needed to construct the route.

Locator File (Practical)

File Number:	MI021D
Data Base Name:	TDBV.PMI021D.P
Number Of Records:	567
File Size:	1 MB
Record Key:	Highest Logical Longitude And Latitude In The Record
Record Format:	Fixed
Record Length:	2036

Record Contents

Each record contains 145 entries which contain the longitude, latitude, internal key and keypoint number (for a keypoint) of a node or keypoint. There is an entry on the file for each node and keypoint in the system. The records are used by the calculation routine to locate the closest node or keypoint to a non-keypoint for developing a calculated distance. The entries are arranged in ascending longitude/latitude sequence.

Time/route File (Practical)

File Number:	MI141B
Data Base Name:	TDBV.MI141B.P
Number Of Records:	18,396
File Size:	76 MB
Record Key: None.	This Is A Relative Record File; The Record Number Is Calculated Using The From And To Key Point Numbers.
Record Format:	Fixed

Record Length: 4088

Record Contents

This file contains timesfrom each keypoint to every other keypoint. It also contains the keypoint strings used to develope the routes between keypoints. Each keypoint except the last (keypoint number 6132) has data contained on six records. Each keypoint except number 1 shares its records with another keypoint. Key Point numbers less than 3067 have their data in the beginning of the records and numbers greater than 3066 have their data in the end of the records. Key Point 2 shares with 6131, 3 shares with 6130 and so on. Thus, while the records are logically variable, the physical organization is fixed.

Surrounding Blocker File (Practical)

File Number: MI306D

Data Base Name: TDBV.PMI306D.P

Number Of Records: 79,921

File Size: 5 MB

Record Key: Internal Key Of The Node To Which The Record Belongs

Record Format: Variable

Record Length: 12-1807

Record Contents

There is one record for each node on the Geographic file. The base portion of the record contains the record key and the number of blocker entries. The blockers must be keypoints. there is an 5 byte entry for each blocker. Each entry contains the blocker keypoint number, travel time from the node to the blocker and a border crossing flag.

Highway Blocker File (Practical)

File Number: MI321D

Number Of Records: 618,531

Data Base Name: TDBV.PMI321D.P

File Size: 58 MB

Record Key: Origin Internal Key, Destination Key Point Number

Record Format: Variable

Record Length: 26-1088

Record Contents

The actual road segments which make up the route from the origin to the destination. The data on the segments includes the highway name, direction and time of travel, ending junction information, and flags for toll roads or bridges. There are also segments containing long names for highways (such as Kennedy Expwy for I94) or alias names.

There is a MI321D record containing the route from each node to each of the blockers for the node. This is why the internal key is used in the key. Nodes have no keypoint numbers. The rest of the records contain keypoint to keypoint routes. There is not a record for each keypoint pair in the system. The route from keypoint A to keypoint B may go through other keypoints. In that case, the MI141B file is used to obtain those keypoints. The MI321D records are then obtained using that list.

Practical Distance Route (A-K-B) File

File Number:	MI341R
Number Of Records:	4,552,979
Data Base Name:	TDBV.PMI341R.P
File Size:	650 MB
Record Key:	Low Internal Key, High Internal Key
Record Format:	Variable
Record Length:	30-1094

Record Contents

Route description between nodes which do not have an intervening keypoint. The data on the variable segments includes the highway name, travel direction and time, ending junction information, and flags for toll roads or bridges. There are also segments containing long names for highways (such as Kennedy expwy for I94) or alias names.

The fixed portion of the record contains the total time needed to travel between the nodes. Unlike the MI340R file, the first and last corporate limit traversals are not included in the record, since they are not used by Practical routing.

For nodes without intervening keypoints, the MI341R file is used in the same manner as the MI321D file. the variable segments supply all of the information needed to construct the route.

Cross Reference File (Practical)

File Number:	MI351D
Number Of Records:	6,132
Data Base Name:	TDBV.PMI351D.P
File Size: .	5 MB
RECORD Key: None.	The File Is A Relative Record File Using The Key Point Number As The Record Number.
Record Format:	Fixed
Record Length:	9

Record Contents

Internal key, keypoint number and keypoint code. This file is used whenever the keypoint number is known and the internal key is needed. It is also used to obtain the keypoint code without reading the Geographic file.

Component File (Practical, Optional)

File Number:	MI391D
Data Base Name:	TDBV.PMI391D.P
Number Of Records:	Variable
File Size:	User Defined
Record Key:	Low Internal Key, Low Key Point Code High Internal Key, High Key Point Code
Record Format:	Variable
Record Length:	16-43

Record Contents

Prestored travel times and route components. When a practical route request is initially made for a given pair of points, a component record is added to the file if space is available. Subsequent requests for that point pair will use the component file to obtain the travel time. This improves response time and saves resources. MileMaker will continue processing normally when MI391D space is exhausted or if the file is not allocated.

There are three record types; the type which is used depends on the keypoint codes of the from and to cities. If one or both of the keypoint codes is a blank, type one is used; if one or both is a 2, type two is used; otherwise, type three is used. Types one and two contain additional information required for calculation points and

secondaries which is not contained on type three records. A component record is not written for routes between two keypoints because the time is obtained by doing one read of the time file.

Locator File (Guide 18 Basic)

File Number:	MI022D
Data Base Name:	TDBV.PMI022D.P
Number Of Records:	389
File Size:	1 MB
Record Key:	Highest Longitude And Latitude In The Record
Record Format:	Fixed
Record Length:	2036

Record Contents

Each record contains 145 entries which contain the longitude, latitude, internal key and keypoint number (for a keypoint) of a node or keypoint. There is an entry on the file for each node and keypoint in the system. The records are used by the calculation routine to locate the closest node or keypoint to a non-keypoint for developing a calculated distance. The entries are arranged in ascending longitude/latitude sequence.

Mileage/Route File (Guide 18 Basic)

File Number:	MI140A
Data Base Name:	TDBV.MI140A.P
Number Of Records:	18,396
File Size:	76 MB
Record Key: None.	Relative Record File. Record Number Is Calculated Using From And To Key Point Numbers.
Record Format:	Fixed
Record Length:	4088

Record Contents

This file contains mileages from each keypoint to every other keypoint. It also contains the keypoint strings used to develop the routes between keypoints. Each keypoint except the last (keypoint number 6132) has data contained on six records. Each keypoint except number 1 shares its records with another keypoint. Key Point

numbers less than 3067 have their data in the beginning of the records and numbers greater than 3067 have their data in the end of the records. Key Point 2 shares with 6131, 3 shares with 6130 and so on. Thus, while the records are logically variable, the physical organization is fixed.

Traversal File (Guide 18 Basic)

File Number:	MI145A
Data Base Name:	TDBV.MI145A.P
Number Of Records:	9,868
File Size: .	1 MB
Record Key:	Low Key Point Number, High Key Point Number
Record Format:	Fixed
Record Length:	8

Record Contents

This file contains traversals between adjacent keypoint pairs on the road network. The term "Adjacent" means that there are no keypoints on the route between the two keypoints. If the route goes thru a keypoint area, one of the blue shaded areas on the Mileage Guide, a record will appear on the traversal file for the keypoint pair. The record contains the first and last traversal keypoint number only. This file is used in conjunction with the mileage/route file to determine whether a rules violation has occurred on a route.

Surrounding Blocker File (Guide 18 Basic)

File Number:	MI307D
Data Base Name:	TDBV.PMI307D.P
Number Of Records:	53,576
File Size:	4 MB
Record Key:	Internal Key Of The Node To Which The Record Belongs
Record Format:	Variable
Record Length:	16-907

Record Contents

There is one record for each node on the geographic file. The base portion of the record contains the record key and the number of blocker entries. The blockers must be keypoints. There is an 9 byte entry for each blocker. Each entry contains the

blocker keypoint number, mileage from the node to the blocker, corporate limit keypoint numbers of keypoint areas traversed, if any, and a border crossing flag.

There may be one or two corporate limit keypoint numbers. These would be for the first and last traversals between the node and the blocker. The blocker is also considered a traversal if it is an HHG keypoint (keypoint code 'K').

Short Distance Mileage File (Guide 18 Basic)

File Number:	MI342M
Data Base Name:	TDBV.PMI342M.P
Number Of Records:	3,337,124
File Size:	22 MB
Record Key:	Low Internal Key, High Internal Key
Record Format:	Variable
Record Length:	16 (See Note Below On Variable Segments)

Record Contents

The distance between nodes which do not have an intervening keypoint, and up to two corporate limit keypoint numbers. This file is needed because if the mileage between such points were derived using the blockers, the mileage would be too high.

Note: the record format is variable for compatability with the MI340R file, but none of the variable segments are used on the MI340M file. Those segments contain the road segments between the nodes. Thus, all records have the same length. The MI340R file can be used in place of the MI340M file by package clients which have routing to conserve disk space.

Cross Reference File (Guide 18 Basic)

File Number:	MI352D
Data Base Name:	TDBV.PMI352D.P
Number Of Records:	6,132
File Size: .	5 MB
Record Key:	None. Relative Record File Using Key Point Number For The Record Number.
Record Format:	Fixed
Record Length:	9

Record Contents

There is a record for each Rand McNally and HHG keypoint. Each record contains the keypoint number, internal key of the keypoint and the keypoint code. These records are used to relate the keypoint number to the internal key. This file saves geographic file reads, which makes the system more efficient.

Secondary Key Point Mileage File (Guide 18 Basic)

File Number:	MI607M
Data Base Name:	TDBV.PMI607M.P
Number Of Records:	Variable
File Size:	User Defined
Record Key:	Internal Key From Geographic File
Record Format:	Variable
Record Length:	33-943

Record Contents

Records on this file may be added by package clients to their file or requested by service clients. File size depends on the number of secondaries desired.

Each record has segments defining the surrounding points for the secondary kepoint. Each segment contains the internal key, keypoint number (if any), longitude and latitude of the surrounding point and, the mileage from the secondary to the surrounding point. The first and last corporate limit keypoints traversed in traveling from secondary to the surrounding point are also included on each segment.

Highway Blocker File (Guide 18, Shortest)

File Number:	MI322D
Data Base Name:	TDBV.PMI322D.P
Number Of Records:	1,157,212
File Size:	50 MB
Record Key:	Origin Internal Key, Destination Key Point Number
Record Format:	Variable
Record Length:	26-980

Record Contents

The actual road segments which make up the route from the origin to the destination. The data on the segments includes the highway name, direction of travel, mileage, ending junction information and flags for toll roads or bridges. There are also segments containing long names for highways (such as Kennedy expwy for I90) or alias names.

There is an MI320D record containing the route from each node to each of the blockers for the node. This is why the internal key is used in the key (nodes have no keypoint numbers). The rest of the records contain keypoint to keypoint routes. There is not a record for each keypoint pair in the system; the route from keypoint A to keypoint B may go through other keypoints. In that case, the MI140A file is used to obtain those keypoints. The MI320D records are then obtained using that list.

Short Distance Route (AKB) File (Guide 18, Shortest)

File Number:	MI342R
Data Base Name:	TDBV.PMI342R.P
Number Of Records:	3,337,126
File Size:	180 MB
Record Key:	Low Internal Key, High Internal Key
Record Format:	Variable
Record Length:	34-988

Record Contents

Route description between nodes not having an intervening keypoint. The base portion of the records is the same as the MI340M file. If a client has routing, he may point the MI340M DD/DLBL statement in the online at the MI340R file. The data on the variable segments includes the highway name, direction of travel, mileage, ending junction information and flags for toll roads or bridges. There are also segments containing long names for highways (such as Kennedy Expwy for I94) or alias names. For nodes without intervening keypoints, the MI340R file is used in the same way as the MI320D file. The variable segments supply all the information needed to construct the route.

Secondary Key Point Route File (Guide 18, Shortest)

File Number:	MI607R
Data Base Name:	TDBV.PMI607R.P
Number Of Records:	Variable
File Size:	User Defined

Record Key:	Internal Key Of Secondary Key Point, Internal Key Of Surrounding Point.
Record Format:	Variable
Record Length:	140-204

Record Contents

This file contains the detailed information for constructing routes from a secondary to its surrounding points. Each record contains up to six road segments detailing the route from the secondary to one of the surrounding points. The data for each segment includes highway name, direction of travel and miles traveled on the segment.

BASIC INSTALLATION COMPONENT LISTS

Appendix

A

Appendix Contents

Module Name	Cics Description (Basic Milemaker)	Module Language	Trans ID	Cics Level	Notes
GE010	Geographic File Inquiry/Maintenance	COBOL	GE10	COMMAND	
GEM010	Map For GE010	BMS-MAP			
GE050	Geographic File Browse	COBOL	GE50	COMMAND	
GEM050	Map For GE050	BMS-MAP			
GE100	Geographic File I/O Access	COBOL		COMMAND	
MI001	MileMaker Master Menu	COBOL	M001	COMMAND	
MIM001	Map For MI001	BMS-MAP			
MI100	Mileage Inquiry	COBOL	M100	COMMAND	
MIM100	Map for MI100	BMS-MAP			
MI103	Origin Mileage Inquiry	COBOL	M103	COMMAND	
MIM103	Map For MI103	BMS-MAP			
MI105	ZIP Code Mileage Processor	COBOL		COMMAND	
MI110	Standard Mileage Interface	COBOL		COMMAND	
MI115	Mileage Processor	COBOL		COMMAND	
MI120	Mileage Calc Between Non-Key Points	COBOL		COMMAND	

Module List (cont.)

Module Name	Cics Description (Basic Milemaker)	Module Language	Trans ID	Cics Level	Notes
MI125	Calculation Routine	COBOL		COMMAND	
MI140	Mileage/Route File Access Module	COBOL		COMMAND	
MI605	Secondary Key Point Inquiry/Update	COBOL	M605	COMMAND	
MIM605	Map For MI605	BMS-MAP			
PM010	Parameter File Inquiry/Update	COBOL	PM10	COMMAND	
PM010MS	Map For PM010	BMS-MAP			
UM010	User Coversion File Maintenance	COBOL	UM10	COMMAND	
UMM010	Map For UM010	BMS-MAP			
UMM011	Map For UM010	BMS-MAP			
MI390	Component File Inquiry/Maintenance	COBOL	M390	COMMAND	
MIM380	Map For MIM390 Browse	BMS-MAP			
MIM390	Map For MI390 Inquiry	BMS-MAP			
MI520	Billing File Write Module	COBOL		COMMAND	
MI9110	Sample On-Line Interface Program	COBOL	MLGC	COMMAND	Working sample of Link to MI110
MIM9110	Map For MI9110	BMS-MAP			

Batch Module List

Module Name	Batch Module Description	Module Language	Notes
MI200	Origin To Destination Mileage Report	COBOL	
MI201	Batch I/O	COBOL	Called by MI200, MI210 load modules.
MI205	ZIP Code Mileage Processor	COBOL	Called by MI200, MI210 load modules.
MI210	Standard Interface	COBOL	Called by user-written program.
MI215	Mileage Processor	COBOL	Called by MI200, MI210 load modules.
MI220	Mileage Calc Between Non-Key Points	COBOL	called by MI200, MI210 load modules.
MI225	Calculation Routine	COBOL	Called by MI200, MI210 load modules.
MI230	Report Generator For Mileage Report	COBOL	Called by MI200, MI210 load modules.
MI240	Mileage/Route File I/O Module	COBOL	Called by MI200, MI210 load modules.
GE101	Geographic File I/O Module	COBOL	Called by MI200, MI210 load modules.
GE965	Geographic File Merge	COBOL	Used in Geographic File installation.
MI921	Key Point And Node Master List	COBOL	
MI951	Geographic File List	COBOL	
MI9390D	Component File Delete	COBOL	
MI9605	Secondary File Conversion	COBOL	
MI9607	Previous Guide Secondary File Conversion	COBOL	
MI9210	Sample Batch Interface Program	COBOL	Working sample of CALL to MI210.
GE966	Geographic Update Program	COBOL	May be used in Geographic file installation.

Current Guide File List

File Name	Current Guide File Description	File Type	File Org	Key Size	Rec Avg	Size Max	Record Count	Meg Size	3390 Disk Space	Notes
GE010DP	Geographic File (Base)	VSAM	KSDS	20	124	124	154282	22	35 CYL	Alternate Keysize: 5. Alternate 2 Keysize: 5. Record count approx *
GE010D1	Geographic File (ALT Index 1)	VSAM	KSDS	5	30	30	Notes	5	8 CYL	Record count same as GE010DP.
GE010D2	Geographic File (ALT Index 2)	VSAM	KSDS	5	30	30	Notes	5	8 CYL	Record count same as GE010DP.
GE020D	Geographic Update File	VSAM	ESDS	NA	146	146	Notes	1	3 CYL	Only used if updating GE010D. Size user-defined.
MI140B	Mileage/Route File	VSAM	RRDS	NA	4088	4088	36792	76	205 CYL	
MI145B	Traversal File	VSAM	KSDS	4	8	8	9868	0.1	1 CYL	
MI015D	3-Digit ZIP Code Mileage File	VSAM	RRDS	NA	2000	2000	916	3	4 CYL	
MI020D	Locator File	VSAM	KSDS	6	2036	2036	389	1	2 CYL	
MI105D	3-Digit ZIP Code Master File	VSAM	KSDS	3	31	31	993	0.1	1 CYL	
MI305D	Surrounding Blocker File	VSAM	KSDS	5	97	2207	51610	5	15 CYL	Record count approx *
MI340M	Short Distance Mileage File	VSAM	KSDS	8	16	18	4823298	53	120 CYL	"Approx 2,494,000 records"
MI350D	Key Point Cross Reference File	VSAM	RRDS	NA	9	9	6132	0.5	1 CYL	
MI390D	Component File	VSAM	KSDS	12	24	42	Notes			Size user-defined
MI520D	Billing File	VSAM	KSDS	19	60	60	Notes			Size user-defined
MI605M	Secondary Key Point Mileage File	VSAM	KSDS	14	267	943	1	0.2	1 CYL	Record count approx *. User may add records.
MI882D	5-Digit Reference File	VSAM	KSDS	8	60	60	150332	4	9 CYL	
PM010D	Parameter File	VSAM	KSDS	18	18	80	100	0.1	1 CYL	Record count approx*. User may add records.

Current Guide File List

File Name	Current Guide File Description	File Type	File Org	Key Size	Rec Avg	Size Max	Record Count	Meg Size	3380 Disk Space	Notes
UM010D	User Conversion File	VSAM	KSDS	25	40	261	16245	0.3	1 CYL	Record count approx *. User may add records.

* See file dump listing provided by TDM for actual record counts.

Previous Guide File List

File Name	Current Guide File Description	File Type	File Org	Key Size	Rec Avg	Size Max	Record Count	Meg Size	3390 Disk Space	Notes
MI140A	Mileage/Route File	VSAM	RRDS	NA	4088	4088	18396	76	124 CYL	
MI145A	Traversal File	VSAM	RRDS	4	8	8	9369	0.1	1 CYL	
MI022D	Locator File	VSAM	KSDS	6	2036	2036	389	1	2 CYL	
MI307D	Surrounding Blocker File	VSAM	KSDS	5	97	1807	53576	4	15 CYL	Record count approx *
MI342M	Short Distance Mileage File	VSAM	KSDS	8	16	16	3337124	40	90 CYL	
MI352D	Key Point Cross Reference File	VSAM	RRDS	NA	9	9	6132	0.5	1 CYL	
MI607M	Secondary Key Point Mileage File	VSAM	KSDS	14	267	943	649	0.2	1 CYL	Record count approx *. User may add records.

* See file dump listing provided by TDM for actual record counts.

MI883D – Same as MI882D

GE011DP – Same as GE010DP

GE011D1 – Same as GE010D1

GE011D2 – Same as GE010D2

Current Guide Dataset Name Worksheet

File Name	Current Guide File Description (Basic)	DDNAME DLBL	TDM's DSN Data Set Names	Define	Client's DSN Data Set Names
GE010D	Geographic File	GE010DP	TDMV.GE010D.PBSE	CLUSTER	
			TDMV.GE010D.PBSEDATA	CLUSTER DATA	
			TDMV.GE010D.PBSEINDX	CLUSTER INDEX	
	Geo File Index Cluster 1		TDMV.GE010D.PALT1	ALT.INDEX1	
			TDMV.GE010D.PALTDAT1	ALT.INDEX DATA1	
			TDMV.GE010D.PALTINX1	ALT.INDEX INDEX1	
GE010D1	Geo File Index Path 1	GE010D1	TDMV.GE010D.PATH1	ALT.INDEX1 PATH	
	Geo File Index Cluster 2		TDMV.GE010D.PALT2	ALT.INDEX 2	
			TDMV.GE010D.PALTDAT2	ALT.INDEX DATA	
			TDMV.GE010D.PALTINX2	ALT.INDEX INDEX2	
GE010D2	Geo File Index Path 2	GE010D2	TDMV.GE010D.PATH2	ALT.INDEX2 PATH	
GE020D	Geographic Update File	GE020D	TDMV.GE020D.PBSE	CLUSTER	
			TDMV.GE020D.PBSEDATA	CLUSTER DATA	
MI140B	Mileage/Route File	MI140B	TDMV.MI140B.PBSE	CLUSTER	
			TDMV.MI140B.PBSEDATA	CLUSTER DATA	
MI145B	Traversal File	MI145B	TDMV.MI145B.PBSE	CLUSTER	
			TDMV.MI145B.PBSEDATA	CLUSTER DATA	
			TDMV.MI145B.PBSEINDX	CLUSTER INDEX	

Current Guide Dataset Name Worksheet (cont.)

File Name	Current Guide File Description (Basic)	DDNAME DLBL	TDM's DSN Data Set Names	Define	Client's DSN Data Set Names
MI015D	3 Digit ZIP Code Mileage File	MI015D	TDMV.MI015D.PBSE	CLUSTER	
			TDMV.MI015D.PBSEDATA	CLUSTER DATA	
			TDMV.MI015D.PBSEINDX	CLUSTER INDEX	
MI020D	Locator File	MI020D	TDMV.MI020D.PBSE	CLUSTER	
			TDMV.MI020D.PBSEDATA	CLUSTER DATA	
			TDMV.MI020D.PBSEINDX	CLUSTER INDEX	
MI105D	3 Digit ZIP Code Master File	MI105D	TDMV.MI105D.PBSE	CLUSTER	
			TDMV.MI105D.PBSEDATA	CLUSTER DATA	
			TDMV.MI105D.PBSEINDX	CLUSTER INDEX	
MI305D	Surrounding Blocker File	MI305D	TDMV.MI305D.PBSE	CLUSTER	
			TDMV.MI305D.PBSEDATA	CLUSTER DATA	
			TDMV.MI305D.PBSEINDX	CLUSTER INDEX	
MI340M	Short Distance Mileage (AKB) File	MI340M	TDMV.MI340M.PBSE	CLUSTER	
			TDMV.MI340M.PBSEDATA	CLUSTER DATA	
			TDMV.MI340M.PBSEINDX	CLUSTER INDEX	
MI350D	Cross Reference File	MI350D	TDMV.MI350D.PBSE	CLUSTER	
			TDMV.MI350D.PBSEDATA	CLUSTER DATA	

Current Guide Dataset Name Worksheet (cont.)

File Name	Current Guide File Description (Basic)	DDNAME DLBL	TDM's DSN Data Set Names	Define	Client's DSN Data Set Names
MI390D	Component File	MI390D	TDMV.MI390D.PBSE	CLUSTER	
			TDMV.MI390D.PBSEDATA	CLUSTER DATA	
			TDMV.MI390D.PBSEINDX	CLUSTER INDEX	
MI520D	Billing File	MI520D	TDMV.MI520D.PBSE	CLUSTER	
			TDMV.MI520D.PBSEDATA	CLUSTER DATA	
			TDMV.MI520D.PBSEINDX	CLUSTER INDEX	
MI605M	Secondary Key Point Mileage File	MI605M	TDMV.MI605M.PBSE	CLUSTER	
			TDMV.MI605M.PBSEDATA	CLUSTER DATA	
			TDMV.MI605M.PBSEINDX	CLUSTER INDEX	
PM010D	Parameter File	PM010D	TDMV.PM010D.PBSE	CLUSTER	
			TDMV.PM010D.PBSEDATA	CLUSTER DATA	
			TDMV.PM010D.PBSEINDX	CLUSTER INDEX	
UM010D	User Conversion File	UM010D	TDMV.UM010D.PBSE	CLUSTER	
			TDMV.UM010D.PBSEDATA	CLUSTER DATA	
			TDMV.UM010D.PBSEINDX	CLUSTER INDEX	
MI882D	5-Digit ZIP Reference File	MI882D	TDMV.MI882D.PBSE	CLUSTER	
			TDMV.MI882D.PBSEDATA	CLUSTER DATA	
			TDMV.MI882D.PBSEINDX	CLUSTER INDEX	

Previous Guide Dataset Name Worksheet

File Name	Current Guide File Description (Basic)	DDNAME DLBL	TDM's DSN Data Set Names	Define	Client's DSN Data Set Names
MI140A	Mileage/Route File	MI140A	TDMV.MI140A.PBSE	CLUSTER	
			TDMV.MI140A.PBSEDATA	CLUSTER DATA	
MI145A	Traversal File	MI145A	TDMV.MI145A.PBSE	CLUSTER	
			TDMV.MI145A.PBSEDATA	CLUSTER DATA	
			TDMV.MI145A.PBSEINDX	CLUSTER INDEX	
MI022D	Locator File	MI022D	TDMV.MI022D.PBSE	CLUSTER	
			TDMV.MI022D.PBSEDATA	CLUSTER DATA	
			TDMV.MI022D.PBSEINDX	CLUSTER INDEX	
MI307D	Surrounding Blocker File	MI307D	TDMV.MI307D.PBSE	CLUSTER	
			TDMV.MI307D.PBSEDATA	CLUSTER DATA	
			TDMV.MI307D.PBSEINDX	CLUSTER INDEX	
MI342M	Short Distance Mileage (AKB) File	MI342M	TDMV.MI342M.PBSE	CLUSTER	
			TDMV.MI342M.PBSEDATA	CLUSTER DATA	
			TDMV.MI342M.PBSEINDX	CLUSTER INDEX	
MI352D	Cross Reference File	MI352D	TDMV.MI352D.PBSE	CLUSTER	
			TDMV.MI352D.PBSEDATA	CLUSTER DATA	
MI607M	Secondary Key Point Mileage File	MI607M	TDMV.MI607M.PBSE	CLUSTER	
			TDMV.MI607M.PBSEDATA	CLUSTER DATA	
			TDMV.MI607M.PBSEINDX	CLUSTER INDEX	

CICS Table List

Table Name	Cics Table Description	TBL Type	Notes
MIDDS	DD STATEMENTS FOR CICS FILES	JCL	Place these DD/DLBL statements in CICS start-up JCL
MIFCT	FCT TABLE ENTRIES	CICS	
MIPCT	PCT TABLE ENTRIES	CICS	
MIPPT	PPT TABLE ENTRIES	CICS	

JCL Name	Batch JCL Description	Notes
GE010DVD	IDCAMS Definition Of Geographic File	Includes definition of 2 altenate index files.
GE020DVD	IDCAMS Definition Of Geographic Update File	
GE965X	Execute Geographic File Merge	
MI015DVD	IDCAMS Definition Of The ZIP Code Mileage File	
MI020DVD	IDCAMS Definition Of Locator File	
MI105DVD	IDCAMS Definition Of ZIP Code MasterFile	
MI140BVD	IDCAMS Definition Of Mileage/Route File	
MI145BVD	IDCAMS Definition Of Traversal File	
MI305DVD	IDCAMS Definition Of Blocker File	
MI340MVD	IDCAMS Definition Of Short Distance Mileage File	
MI350DVD	IDCAMS Definition Of Cross Reference File	
MI390DVD	IDCAMS Definition Of Component File	
MI520DVD	IDCAMS Definition Of Billing File	
MI605MVD	IDCAMS Definition Of Secondary Key Point Mileage File	
MI605MVR	Load TDM Records Into MI605M File	IDCAMS repro with replace option.
MI9605X	Convert User Secondary record	File Define file and load dummy.
PM010DVD	IDCAMS Definition Of Parameter File	
PM010DVR	Load TDM Records Into Parameter File	IDCAMS repro with replace option.
UM010DVD	IDCAMS Definition Of User Conversion File	
MI882DVD	ZIP5 Reference File Load	

Batch JCL List (cont.)

JCL Name	Batch JCL Description	Notes
UM010DVR	Load TDM Records Into UC010D File	IDCAMS repro with replace option.
MI200X	Execute JCL Mileage Inquiry Report	
MI200L	Sample Link For MI200	Provided in case relink of the MI200 load module is required.
MI210L	Sample Link For MI210	Provided in case relink of the MI210 load module is required.
MI210X	JCL Statements For Mileage Interface	
MI921X	Execute JCL For Key Point And Node Master List	
MI9390DX	Execute JCL For Component File Delete	
MI951X	Execute JCL For Geographic File List	
MI9210X	Execute JCL For Sample Batch Interface	
MI022DVD	IDCAMS Definition Of Locator File	Previous Guide File
MI140AVD	IDCAMS Definition Of Mileage/Route File	Previous Guide File
MI145AVD	IDCAMS Definition Of Traversal File	Previous Guide File
MI307DVD	IDCAMS Definition Of Blocker File	Previous Guide File
MI342MVD	IDCAMS Definition Of Short Distance Mileage File	Previous Guide File
MI352DVD	IDCAMS Definition Of Cross Reference File	Previous Guide File
MI607MVD	IDCAMS Definition Of Secondary Key Point Mileage File	Previous Guide File
MI607MVR	Load TDM Records Into MI607M File	IDCAMS repro with replace option.
MI9607X	Convert User Secondary File	Creates Previous Guide File.
MI607MVX	IDCAM's Load Of Previous Guide Secondary Key Point Mileage File Using Client's Data	The client's data file is created by MI9605X.
GE966X	Executes Geographic file update	

SHORTEST DISTANCE ROUTING COMPONENT LISTS

Appendix

B

Appendix Contents

Module List

Module Name	CICS Module Description	Module Language	Trans ID	CICS Level	Notes
MI310	Route Inquiry	COBOL	MI31	COMMAND	
MI310MS	Map For MI310	BMS-MAP			
MI320	Route Display	COBOL	MI32	COMMAND	
MI320MS	Map For MI320	BMS-MAP			
MI335	Online Route Interface Module	COBOL		COMMAND	
MI606	Secondary Route File Maintenance	COBOL	M606	COMMAND	
MI9335	Sample Online Interface Module	COBOL	MRTC	COMMAND	Working sample of link to MI335.
MIM9335	Map For MI9335	BMS-MAP			
MI7335	Online Route Interface with Routes	COBOL		COMMAND	
MI79335	Sample Online Interface	COBOL	MRTR	COMMAND	Working sample of link to MI7335.
MIM7335	Map for MI79335	BMS-MAP			

Batch Module List

Module Name	Batch Module Description	Module Language	Notes
GE101	Batch Geographic File Access	COBOL	Basic MileMaker module required for route processing.
GE102	Batch Previous Guide Geographic File Access	COBOL	Basic MileMaker module required for route processing.
MI201	Batch I/O	COBOL	Provides access to Basic MileMaker and route files.
MI215	Mileage Processor	COBOL	Called by MI420 to obtain route components.
MI220	Mileage Calc Between Non-key Points	COBOL	Basic MileMaker module required for route processing.
MI225	Calculation Routine	COBOL	Basic MileMaker module required for route processing.
MI240	Mileage/route File I/O Module	COBOL	Basic MileMaker module required for route processing.
MI410	Batch Route Inquiry	COBOL	
MI420	Batch Route Module	COBOL	
MI430	Batch Route Print Module	COBOL	
MI435	Batch Route Interface Module	COBOL	
MI9435	Sample Batch Route Interface Module	COBOL	"Working sample of ""call"" to MI435."
MI7420	Batch Route Module	COBOL	Called by MI7435
MI7435	Batch Route Interface	COBOL	
MI79435	Sample Batch Route Interface Module	COBOL	"Working sample of ""call"" to MI7435."

File List

File Name	File Description	File Type	File Org	Key Size	Rec Avg	Size Max	Record Count	Meg Size	3380 Disk Space	Notes
MI320D	Highway Blocker File	VSAM	KSDS	6	90	1088	1394619	124	280 CYL	Record count approx *
MI340R	Shortest Route (A-K-B) File	VSAM	KSDS	8	116	2318	4823298	344	1277 CYL	Approx 2,500,000 records*
MI605R	Secondary Key Point Route File	VSAM	KSDS	10	140	204	1	0.5	1 CYL	Record count approx *
MI322D	Highway Blocker File (prev.)	VSAM	KSDS	6	105	980	1157212	66	215 CYL	Record count approx *
MI342R	Shortest Route (A-K-B) File (prev.)	VSAM	KSDS	8	116	2086	3337126	353	848 CYL	
MI607R	Secondary Key Point Route File (prev.)	VSAM	KSDS	10	140	204	1351	0.5	1 CYL	Record count approx *

* See file dump listing provided by TDM for actual record counts.

File Dataset Name Worksheet

File Name	File Description	DDNAME DLBL	TDM's DSN Data Set Names	Client's Define	DSN Data Set Names
MI320D	Highway Blocker File	MI320D	TDMV.MI320D.PBSE	CLUSTER	
			TDMV.MI320D.PBSEDATA	CLUSTER DATA	
			TDMV.MI320D.PBSEINDX	CLUSTER INDEX	
MI340R	Shortest Route (A-K-B) File		TDMV.MI340R.PBSE	CLUSTER	
			TDMV.MI340R.PBSEDATA	CLUSTER DATA	
			TDMV.MI340R.PBSEINDX	CLUSTER INDEX	
MI605R	Secondary Key Point Route	MI605R	TDMV.MI605R.PBSE	CLUSTER	
			TDMV.MI605R.PBSEDATA	CLUSTER DATA	
			TDM.MI605R.PBSEINDX	CLUSTER INDEX	
MI322D	Highway Blocker File (previous)	MI322D	TDMV.MI322D.PBSE	CLUSTER	
			TDMV.MI322D.PBSEDATA	CLUSTER DATA	
			TDMV.MI322D.PBSEINDX	CLUSTER INDEX	
MI342R	Shortest Route (A-K-B) File (previous)	MI342R	TDMV.MI342R.PBSE	CLUSTER	
			TDMV.MI342R.PBSEDATA	CLUSTER DATA	
			TDMV.MI342R.PBSEINDX	CLUSTER INDEX	
MI607R	Secondary Key Point Route (previous)	MI607R	TDMV.MI607R.PBSE	CLUSTER	
			TDMV.MI607R.PBSEDATA	CLUSTER DATA	
			TDMV.MI607R.PBSEINDX	CLUSTER INDEX	

CICS Table List

Table Name	CICS Table Description	TBL Type	Notes
MRDDS	DD STATEMENTS FOR CICS FILES	JCL	Add these DD statements in CICS start-up JCL
MRFCT	FCT TABLE ENTRIES	CICS	
MRPCT	PCT TABLE ENTRIES	CICS	
MRPPT	PPT TABLE ENTRIES	CICS	

JCL Name	JCL Description	Notes
MI320DVD	IDCAMS Definition Of Highway Blocker File	
MI340RVD	IDCAMS Definition Of Short Distance Route File	
MI605RVD	IDCAMS Definition Of Secondary Key Point Route File	Define and load dummy record.
MI605RVR	IDCAMS Load Of Secondary Key Point Route File	Load Rand McNally data from Distribution tape.
MI322DVD	IDCAMS Definition Of Highway Blocker File	
MI342RVD	IDCAMS Definition Of Short Distance Route File	
MI607RVD	IDCAMS Definition Of Previous Guide Secondary Key Point Route File	Define and load dummy record.
MI607RVR	IDCAMS Load Of Previous Guide Secondary Key Point Route File	Load Rand McNally data from Distribution tape.
MI410X	Execute Batch Shortest Routing	
MI410L	Sample Link For Batch Report System	
MI435X	JCL Needed To Use Batch Interface	
MI435L	Sample Link Of Batch Interface	
MI607RVX	IDCAMS Load Of Previous Guide Secondary Key Point Route file using client's data	The client's file is created by MI9605X.
MI9435X	Execute Sample Batch Interface	
MI7435X	JCL Needed to Use Batch Interface	
MI79435X	Execute Sample Batch Interface	

PRACTICAL ROUTE COMPONENT LISTS

Appendix C

Appendix Contents

Module List

Module Name	CICS Module Description	Module Language	Trans ID	CICS Level	Notes
MI310	Route Inquiry Input Map Processor	COBOL	MI31	COMMAND	
MI310MS	Map For MI310	BMS-MAP			
MI315	Practical Mileage Display	COBOL	MI35	COMMAND	
MI315MS	Map For MI315	BMS-MAP			
MI320	Route Display	COBOL	MI32	COMMAND	
MI320MS	Map For MI320	BMS-MAP			
MI335	Online Route Interface Module	COBOL		COMMAND	
MI606	Secondary Route File Maintenance	COBOL	M606	COMMAND	
MI9335	Sample Online Interface Module	COBOL	MRTC	COMMAND	Working sample of link to MI335.
MIM9335	Map For MI9335	BMS-MAP			
MI7335	Online Route Interface with Routes	COBOL		COMMAND	
MI79335	Sample Online Interface	COBOL	MRTR	COMMAND	Working sample of link to MI7335.
MIM7335	Map for MI79335	BMS-MAP			

Batch Module List

Module Name	Batch Module Description	Module Language	Notes
GE101	Batch Geographic File Access	COBOL	Basic MileMaker module required for route processing.
MI201	Batch I/O	COBOL	Provides access to Basic MileMaker and route files.
MI215	Mileage Processor	COBOL	Called by MI420 to obtain route components.
MI220	Mileage Calc Between Non-key Points	COBOL	Basic MileMaker module required for route processing.
MI225	Calculation Routine	COBOL	Basic MileMaker module required for route processing.
MI240	Time/Route File I/O Module	COBOL	Basic MileMaker module required for route processing.
MI410	Batch Route Inquiry	COBOL	
MI420	Batch Route Module	COBOL	
MI430	Batch Route Print Module	COBOL	
MI435	Batch Route Interface Module	COBOL	
MI9435	Sample Batch Route Interface Module	COBOL	"Working sample of ""call"" to MI435."
MI7420	Batch Route Module	COBOL	Called by MI7435
MI7435	Batch Route Interface	COBOL	
MI79435	Sample Batch Route Interface Module	COBOL	"Working sample of ""call"" to MI7435."

File List

File Name	File Description	File Type	File Org	Key Size	Rec Avg	Size Max	Record Count	Meg Size	3390 Disk Space	Notes
MI321D	Highway Blocker File	VSAM	KSDS	6	70	1088	618531	32	90 CYL	Record count approx *
MI341R	Practical Route (A-K-B) File	VSAM	KSDS	8	90	1094	NOTES	238	810 CYL	Approx 4,552,979 records*
MI141B	Time/Route File	VSAM	RRDS	NA	4088	4088	18396	76	124 CYL	
MI306D	Surrounding Blocker File	VSAM	KSDS	5	44	1807	79921	2	10 CYL	Record count approx *
MI021D	Practical Locator File	VSAM	KSDS	6	2036	2036	558	1	2 CYL	
MI351D	Practical Cross Reference File	VSAM	RRDS	NA	9	9	6132	0.5	1 CYL	
MI391D	Practical Route Component File	VSAM	KSDS	12	24	42				Size is user-dependent. See Chapter 8.
MI605R	Secondary Key Point Route File	VSAM	KSDS	10	140	204	1	0.5	1 CYL	Record count approx *

* See file dump listing provided by TDM for actual record counts.

File Dataset Name Worksheet

File Name	File Description	DDNAME DLBL	TDM's DSN Data Set Names	Client's Define	DSN Data Set Names
MI321D	Highway Blocker File	MI321D	TDMV.MI321D.PBSE	CLUSTER	
			TDMV.MI321D.PBSEDATA	CLUSTER DATA	
			TDMV.MI321D.PBSEINDX	CLUSTER INDEX	
MI341R	Practical Route (A-K-B) File		TDMV.MI341R.PBSE	CLUSTER	
			TDMV.MI341R.PBSEDATA	CLUSTER DATA	
			TDMV.MI341R.PBSEINDX	CLUSTER INDEX	
MI141B	Time/Route File	MI141B	TDMV.MI141B.PBSE	CLUSTER	
			TDMV.MI141B.PBSEDATA	CLUSTER DATA	
MI021D	Practical Locator File	MI021D	TDMV.MI021D.PBSE	CLUSTER	
			TDMV.MI021D.PBSEDATA	CLUSTER DATA	
			TDM.MI021D.PBSEINDX	CLUSTER INDEX	
MI306D	Practical Surrounding Blocker File	MI306D	TDMV.MI306D.PBSE	CLUSTER	
			TDMV.MI306D.PBSEDATA	CLUSTER DATA	
			TDMV.MI306D.PBSEINDX	CLUSTER INDEX	
MI351D	Practical Cross Reference File	MI351D	TDMV.MI351D.PBSE	CLUSTER	
			TDMV.MI351D.PBSEDATA	CLUSTER DATA	
MI391D	Practical Route Component File	MI391D	TDMV.MI391D.PBSE	CLUSTER	
			TDMV.MI391D.PBSEDATA	CLUSTER DATA	
			TDMV.MI391D.PBSEINDX	CLUSTER INDEX	
MI605R	Secondary Key Point Route File	MI605R	TDMV.MI605R.PBSE	CLUSTER	
			TDMV.MI605R.PBSEDATA	CLUSTER DATA	
			TDMV.MI605R.PBSEINDX	CLUSTER INDEX	

CICS Table List

Table Name	CICS Table Description	TBL Type	Notes
PRDDS	DD STATEMENTS FOR FILES	JCL	Add these DD statements in CICS start-up JCL
PRFCT	FCT TABLE ENTRIES	CICS	
PRPCT	PCT TABLE ENTRIES	CICS	
PRPPT	PPT TABLE ENTRIES	CICS	

JCL Name	JCL Description	Notes
MI321DVD	IDCAMS Definition Of Highway Blocker File	
MI341RVD	IDCAMS Definition Of Practical Route File	
MI141BVD	IDCAMS Definition Of Time/Route File	
MI306DVD	IDCAMS Definition Of Surrounding Blocker File	
MI351DVD	IDCAMS Definition Of Key Point Cross Reference File	
MI021DVD	IDCAMS Definition Of Locator File	
MI391DVD	IDCAMS Definition Of Optional Component File	
MI605RVD	IDCAMS Definition Of Secondary Key Point Route File	Define and load dummy record.
MI605RVR	IDCAMS Load Of Secondary Key Point Route File	Load Rand McNally data from Distribution tape.
MI410X	Route Edit	
MI435X	JCL Statements For Route Interface	
MI410L	Sample Link Of Batch Route Report	
MI435L	Sample Link Of Route Interface	
MI9435X	Execute Sample Batch Interface	
MI7435X	JCL Needed to Use Batch Interface	
MI79435X	Execute Sample Batch Interface	

MILEAGE/ZIP INTERFACE SAMPLES

Appendix

D

Appendix Contents

Layout of Storage Area (online)

```
01 MI-INTERFACE-AREA
     05 MI-POINT-DATA
           10 MI-REQUEST-TYPE                              PIC X(1).
           10 MI-MILEAGE-IND                               PIC X(1).
           10 MI-ERROR-TYPE                                PIC X(2).
SPACE=NO ERROR
NF=AT LEAST ONE POINT WAS NOT FOUND
NP=PREVIOUS GUIDE NOT SUPPORTED
NE=NUMBER OF ENTRIES INVALID
ST=AT LEAST ONE ENTRY ENCOUNTERED A MILEAGE ERROR
CE=A FILE ERROR HAS OCCURED
           10 MI-NBR-ENTRIES                               PIC 9(2).
           10 MI-NBR-ENTRIES-X REDEFINES MI-NBR-ENTRIES.   PIC X(2).
           10 MI-ENTRY-POINTS OCCURS 40 TIMES.
              15 MI-STATE-CITY.
                    20 MI-STATE                            PIC X(2).
                    20 MI-CITY.
                       25 MI-CY                            PIC X(15).
                       25 MI-CTY                           PIC X(3).
                       20 FILLER                           PIC X(17).
              15 MI-ZIP5                                   PIC X(05).
              15 MI-SPLC                                   PIC 9(9)    COMP-3.
              15 MI-SPLCX REDEFINES MI-SPLC                PIC X(5).
              15 MI-ERROR-CODE                             PIC X(2).
              15 MI-ERROR-FILE                             PIC X(7).
              15 MI-STATUS                                 PIC X(1).
              15 MI-MILEAGE                                PIC S9(5)   COMP-3.
              15 FILLER OCCURS 2 TIMES.
                    20 MI-AUD-INTERNAL-KEY                 PIC 9(9)    COMP-3.
                    20 MI-AUD-MILES                        PIC S9(3)   COMP-3.
01 AC-CLIENT-FIELDS**
     05 AC-CLIENT-ID                                       PIC X(04).
```

** Four bytes area for batch programs. (Do not include for online programs.)

Online Access Sample Program

```
IDENTIFICATION DIVISION.
PROGRAM-ID. USERPGM.
AUTHOR. RAND MCNALLY & COMPANY.
REMARKS.    THIS IS A SAMPLE COMMAND LEVEL PROGRAM ALLOWING
            ENTRY TO THE MILEMAKER SCREENS MIM001, MIM100,
            MIM103, AND MIM310. AFTER MILEMAKER PROCESSING
            IS COMPLETE MILEMAKER WILL XCTL TO THIS PROGRAM
            AND RETURN EIBCALEN = 132.
      EJECT
ENVIRONMENT DIVISION.
DATA DIVISION.
WORKING-STORAGE SECTION.
01 USER-COMM-AREA
05 FILLER                                          PIC X(??).

01 PROGRAM-CONSTANTS.
      05 THIS-MODULE-NBR                           PIC X(08)VALUE 'USERPGM'.
      05 MENU-MODULE                               PIC X(08) VALUE 'MI001'.

01 LENGTHS
05 MI-TWA-LENGTH                                  PIC S9(04) COMP VALUE +132.

01 TEST-USER-REQUEST.
      05 MILEMAKER-SCREEN                          PIC X(01).
            88 MILEMAKER-MIM001                    VALUE ' '.
            88 MILEMAKER-MIM100                    VALUE ' '.
            88 MILEMAKER-MIM103                    VALUE ' '.
            88 MILEMAKER-MIM310                    VALUE ' '.

01 TEST-VALID-USER-ID
      05 USER-ID-1                                 PIC X(04).
            88 VALID-ID                            VALUE 'TDM ', ' ',
                                  ...
                                  ...
                                  ...
01 TWA-WORK-AREA SYNC.
      02 TWA-AREA.
            05 TWA-PROGRAM-NAME.
               10 TWA-PROG-NAME-7                  PIC X(07).
               10 TWA-PROG-NAME-1                  PIC X(01).
```

```
            05 TWA-CONTROL-PARMS.
               10 TWA-SECURITY-CODE                PIC X(01).
               10 TWA-XCTL-IND                     PIC X(01).
                     88 TWA-XCTL-MI100             VALUE '1'.
                     88 TWA-XCTL-MI103             VALUE '2'.
                     88 TWA-XCTL-MI310             VALUE '8'.
               10 FILLER                           PIC X(02).
            05 TWA-CLIENT-CODE                     PIC X(04).
            05 TWA-PASSWORD                        PIC X(08).
            05 TWA-MILEAGE-GUIDE                   PIC X(01).
            05 FILLER                              PIC X(03).
            05 TWA-ADDRESS-NBR-1                   PIC S9(08) COMP.
            05 FILLER                              PIC X(100).

*01 DFHBMSCA COPY DFHBMSCA.
      COPY DFHBMSCA.
*01 DFHAID COPY DFHAID.
      COPY DFHAID.
LINKAGE SECTION.
01 DFHCOMMAREA.
            05 DA-AREA PIC X(132).
            SKIP3
PROCEDURE DIVISION.
*
* RETURN FROM MILEMAKER MODULE MI001
*
      IF EIBCALEN = MI-TWA-LENGTH
            MOVE DFHCOMMAREA TO TWA-AREA
            PERFORM USER-RETURN-FROM-MI001-MODULE-PROCEDURE.
*
* TRANSFER CONTROL TO MILEMAKER MODULE MI001
*
      MOVE LOW-VALUE TO TWA-WORK-AREA.
      MOVE THIS-MODULE-NBR TO TWA-PROGRAM-NAME.
      MOVE USER-ID-1 TO TWA-CLIENT-CODE.

      IF MILEMAKER-MIM001
            MOVE ' ' TO TWA-XCTL-IND
            EXEC CICS
               XCTL PROGRAM (MENU-MODULE)
                     COMMAREA (TWA-AREA)
                     LENGTH (MI-TWA-LENGTH)
            END-EXEC.

      IF MILEMAKER-MIM100
```

```
      MOVE '1' TO TWA-XCTL-IND
      EXEC CICS
         XCTL PROGRAM (MENU MODULE)
         COMMAREA (TWA-AREA)
         LENGTH (MI-TWA-LENGTH)
      END-EXEC.

IF MILEMAKER-MIM103
      MOVE '2' TO TWA-XCTL-IND
      EXEC CICS
         XCTL PROGRAM (MENU MODULE)
         COMMAREA (TWA-AREA)
         LENGTH (MI-TWA-LENGTH)
      END-EXEC.

IF MILEMAKER-MIM310
      MOVE '8' TO TWA-XCTL-IND
      EXEC CICS
      XCTL PROGRAM (MENU MODULE)
         COMMAREA (TWA-AREA)
         LENGTH (MI-TWA-LENGTH)
      END-EXEC.
```

Working Sample Program - Command Level CICS

Refer to member MI9110 on dataset TDM.SRC.T on the system distribution tape for a working sample command level CICS interface program.

Working Sample Program - Batch

Refer to member MI9210 on dataset TDM.SRC.T on system distribution tape for a working sample batch interface program. Use member MI9210X on dataset TDM.JCL.T to execute MI9210.

PRACTICAL & SHORTEST ROUTING INTERFACE SAMPLES

Appendix

E

Appendix Contents

Interface Area Cobol Definition

```
02 MI - INTERFACE AREA.
     05 MI-POINT-DATA.
          10 MI-REQUEST-TYPE                              PIC X(1).
          10 MI-MILEAGE-IND                               PIC X(1).
          10 MI-ERROR-TYPE                                PIC X(2).
SPACE = NO ERROR
NF = AT LEAST ONE POINT WAS NOT FOUND
NP = PREVIOUS GUIDE NOT SUPPORTED
NS = SHORTEST ROUTES NOT SUPPORTED
NT = PRACTICAL ROUTES NOT SUPPORTED
NE = NUMBER OF ENTRIES INVALID
CE = A FILE ERROR HAS OCCURRED
ST = AT LEAST ONE ENTRY ENCOUNTERED A MILEAGE ERROR
          10 MI-NBR-ENTRIES                              PIC 9(2).
          10 MI-NBR-ENTRIES-X REDEFINES MI-NBR-ENTRIES.   PIC X(2).
          10 MI-ENTRY-POINTS OCCURS 40 TIMES.
               15 MI-STATE-CITY.
                    20 MI-STATE                           PIC X(2).
                    20 MI-CITY.
                        25 MI-CY                          PIC X(15).
                        25 MI-CTY                         PIC X(3).
                    20 FILLER                             PIC X(17).
               15 MI-ZIP5                                 PIC X(05)
               15 MI-SPLC                                 PIC 9(9)    COMP-3.
               15 MI-SPLCX REDEFINES MI-SPLC              PIC X(5).
               15 MI-ERROR-CODE                           PIC X(2).
               15 MI-ERROR-FILE                           PIC X(7).
          10 MI-ROUTE-DATA.
               15 MI-CALLING-PROGRAM                      PIC X(08).
               15 MI-TERM-ID                              PIC X(04).
               15 MI-RTE-STATUS                           PIC X(01).
               15 MI-RTE-ERROR-MSG                        PIC X(51).
               15 MI-RTE-TOTAL-MILEAGE                    PIC S9(05)  COMP-3.
               15 MI-RTE-TOLL-MILEAGE                     PIC S9(05)  COMP-3.
               15 MI-TS-QUE-NAME                          PIC X(08).
               15 MI-TS-NUM-PAGES                         PIC S9(03)  COMP-3.
               15 MI-ST-MLG-TABLE-ADDR                    PIC S9(08)  COMP.
```

State Mileage Table Cobol Definition

```
02 STATE-MILEAGE-TABLE.
   05 ST-TABLE.
      10 ST-RTE-TYPE                                     PIC X(01).
```

```
   10 ST-COUNT                                          PIC S9(03)  COMP-3.
   10 ST-MILEAGES OCCURS 98 TIMES.
       15 ST-MI                                         PIC S9(05)  COMP-3.
       15 ST-TRIP-MILES.
            20 ST-MI-LOADED                             PIC S9(05)  COMP-3.
            20 ST-MI-EMPTY                              PIC S9(05)  COMP-3.
            20 ST-MI-TOLL                               PIC S9(05)  COMP-3.
            20 ST-MI-BOBTAIL                            PIC S9(05)  COMP-3.
05 FILLER                                               PIC X(1202).
```

State Table List

```
01 STATE-TABLE
   05 FILLER                                                PIC X(25)
              VALUE 'AL ALABAMA '.
   05 FILLER                                                PIC X(25)
              VALUE 'AK ALASKA '.
   05 FILLER                                                PIC X(25)
              VALUE 'AZ ARIZONA '.
   05 FILLER                                                PIC X(25)
              VALUE 'AR ARKANSAS '.
   05 FILLER                                                PIC X(25)
              VALUE 'CA CALIFORNIA '.
   05 FILLER                                                PIC X(25)
              VALUE 'CO COLORADO '.
   05 FILLER                                                PIC X(25)
              VALUE 'CT CONNECTICUT '.
   05 FILLER                                                PIC X(25)
              VALUE 'DE DELEWARE '.
   05 FILLER                                                PIC X(25)
              VALUE 'DC DISTRICT OF COLUMBIA '.
   05 FILLER                                                PIC X(25)
              VALUE 'FL FLORIDA '.
   05 FILLER                                                PIC X(25)
              VALUE 'GA GEORGIA '.
   05 FILLER                                                PIC X(25)
              VALUE 'HI HAWAII '.
   05 FILLER                                                PIC X(25)
              VALUE 'ID IDAHO '.
   05 FILLER                                                PIC X(25)
              VALUE 'IL ILLINOIS '.
   05 FILLER                                                PIC X(25)
              VALUE 'IN INDIANA '.
   05 FILLER                                                PIC X(25)
              VALUE 'IA IOWA '.
   05 FILLER                                                PIC X(25)
              VALUE 'KS KANSAS '.
   05 FILLER                                                PIC X(25)
              VALUE 'KY KENTUCKY '.
   05 FILLER                                                PIC X(25)
              VALUE 'LA LOUISIANA '.
   05 FILLER                                                PIC X(25)
```

```
            VALUE 'ME MAINE '.
05 FILLER                                                PIC X(25)
            VALUE 'MD MARYLAND '.
05 FILLER                                                PIC X(25)
            VALUE 'MA MASSACHUSETTS '.
05 FILLER                                                PIC X(25)
            VALUE 'MI MICHIGAN '.
05 FILLER                                                PIC X(25)
            VALUE 'MN MINNESOTA '.
05 FILLER                                                PIC X(25)
            VALUE 'MS MISSISSIPPI '.
05 FILLER                                                PIC X(25)
            VALUE 'MO MISSOURI '.
05 FILLER                                                PIC X(25)
            VALUE 'MT MONTANA '.
05 FILLER                                                PIC X(25)
            VALUE 'NE NEBRASKA '.
05 FILLER                                                PIC X(25)
            VALUE 'NV NEVADA '.
05 FILLER                                                PIC X(25)
            VALUE 'NH NEW HAMPSHIRE '.
05 FILLER                                                PIC X(25)
            VALUE 'NJ NEW JERSEY '.
05 FILLER                                                PIC X(25)
            VALUE 'NM NEW MEXICO '.
05 FILLER                                                PIC X(25)
            VALUE 'NY NEW YORK '.
05 FILLER                                                PIC X(25)
            VALUE 'NC NORTH CAROLINA '.
05 FILLER                                                PIC X(25)
            VALUE 'ND NORTH DAKOTA '.
05 FILLER                                                PIC X(25)
            VALUE 'OH OHIO '.
05 FILLER                                                PIC X(25)
            VALUE 'OK OKLAHOMA '.
05 FILLER                                                PIC X(25)
            VALUE 'OR OREGON '.
05 FILLER                                                PIC X(25)
            VALUE 'PA PENNSYLVANIA '.
05 FILLER                                                PIC X(25)
            VALUE 'RI RHODE ISLAND '.
05 FILLER                                                PIC X(25)
            VALUE 'SC SOUTH CAROLINA '.
05 FILLER                                                PIC X(25)
            VALUE 'SD SOUTH DAKOTA '.
```

```
05 FILLER                                                PIC X(25)
           VALUE 'TN TENNESSEE '.
05 FILLER                                                PIC X(25)
           VALUE 'TX TEXAS '.
05 FILLER                                                PIC X(25)
           VALUE 'UT UTAH '.
05 FILLER                                                PIC X(25)
           VALUE 'VT VERMONT '.
05 FILLER                                                PIC X(25)
           VALUE 'VA VIRGINIA '.
05 FILLER                                                PIC X(25)
           VALUE 'WA WASHINGTON '.
05 FILLER                                                PIC X(25)
           VALUE 'WV WEST VIRGINIA '.
05 FILLER                                                PIC X(25)
           VALUE 'WI WISCONSIN '.
05 FILLER                                                PIC X(25)
           VALUE 'WY WYOMING '.
05 FILLER                                                PIC X(25)
           VALUE 'AB ALBERTA '.
05 FILLER                                                PIC X(25)
           VALUE 'BC BRITISH COLUMBIA '.
05 FILLER                                                PIC X(25)
           VALUE 'MB MANITOBA '.
05 FILLER                                                PIC X(25)
           VALUE 'NB NEW BRUNSWICK '.
05 FILLER                                                PIC X(25)
           VALUE 'NF NEWFOUNDLAND '.
05 FILLER                                                PIC X(25)
           VALUE 'NT NW TERRITORIES '.
05 FILLER                                                PIC X(25)
           VALUE 'NS NOVA SCOTIA '.
05 FILLER                                                PIC X(25)
           VALUE 'ON ONTARIO '.
05 FILLER                                                PIC X(25)
           VALUE 'PE PRINCE EDWARD ISLAND '.
05 FILLER                                                PIC X(25)
           VALUE 'QC QUEBEC '.
05 FILLER                                                PIC X(25)
           VALUE 'SK SASKATCHEWAN '.
05 FILLER                                                PIC X(25)
           VALUE 'YT YUKON TERRITORY '.
05 FILLER                                                PIC X(25)
           VALUE 'PR PUERTO RICO '.
05 FILLER                                                PIC X(25)
```

```
            VALUE 'AG AGUASCALIENTES '.
05 FILLER                                                PIC X(25)
            VALUE 'BJ BAJA CALIFORNIA '.
05 FILLER                                                PIC X(25)
            VALUE 'BS BAJA CALIFORNIA-SUR '.
05 FILLER                                                PIC X(25)
            VALUE 'CP CAMPECHE '.
05 FILLER                                                PIC X(25)
            VALUE 'CH CHIAPES '.
05 FILLER                                                PIC X(25)
            VALUE 'CI CHIHUAHUA '.
05 FILLER                                                PIC X(25)
            VALUE 'CU COAHUILA DE ZARGOZA '.
05 FILLER                                                PIC X(25)
            VALUE 'CL COLIMA '.
05 FILLER                                                PIC X(25)
            VALUE 'DF DISTRITO FEDERAL '.
05 FILLER                                                PIC X(25)
            VALUE 'DG DURANGO '.
05 FILLER                                                PIC X(25)
            VALUE 'EM ESTADO MEXICO '.
05 FILLER                                                PIC X(25)
            VALUE 'GJ GUANAJUATO '.
05 FILLER                                                PIC X(25)
            VALUE 'GR GUERRERO '.
05 FILLER                                                PIC X(25)
            VALUE 'HG HIDALGO '.
05 FILLER                                                PIC X(25)
            VALUE 'JA JALISCO '.
05 FILLER                                                PIC X(25)
            VALUE 'MH MICHOACAN '.
05 FILLER                                                PIC X(25)
            VALUE 'MR MORELOS '.
05 FILLER                                                PIC X(25)
            VALUE 'NA NAYARIT '.
05 FILLER                                                PIC X(25)
            VALUE 'NL NUEVO LEON '.
05 FILLER                                                PIC X(25)
            VALUE 'OA OAXACA '.
05 FILLER                                                PIC X(25)
            VALUE 'PU PUEBLA '.
05 FILLER                                                PIC X(25)
            VALUE 'QA QUERETARO '.
05 FILLER                                                PIC X(25)
            VALUE 'QR QUINTANA ROO '.
```

```
   05 FILLER                                                PIC X(25)
              VALUE 'SL SAN LUIS POTOSI '.
   05 FILLER                                                PIC X(25)
              VALUE 'SI SINALOA '.
   05 FILLER                                                PIC X(25)
              VALUE 'SO SONORA '.
   05 FILLER                                                PIC X(25)
              VALUE 'TA TABASCO '.
   05 FILLER                                                PIC X(25)
              VALUE 'TM TAMAULIPAS '.
   05 FILLER                                                PIC X(25)
              VALUE 'TL TLAXCALA '.
   05 FILLER                                                PIC X(25)
              VALUE 'VL VERA CRUZ-LLAVE '.
   05 FILLER                                                PIC X(25)
              VALUE 'YC YUCATAN '.
   05 FILLER                                                PIC X(25)
              VALUE 'ZT ZACATECAS '.
05 FILLER                                                PIC X(25)
              VALUE 'NU NUNAVUT '.

05 FILLER                                                PIC X(25)
              VALUE 'XX XXXXXXXXXXXXXXXXXX '.

01 STATE-TABLE-R   REDEDINES STATE-TABLE
   05 STATE ENTRY OCCURS 98 TIMES
      10 ST - ABBREV                                        PIC X(02).
      10 FILLER                                             PIC X(01).
      10 ST - NAME                                          PIC X(22).
```

Route Segment Table - Batch

```
05 ROUTE-INTERFACE-SEGMENT-TABLE.
   07 ROUTE-INTERFACE-RTE-TABLE-AREA.
      10 RI2-TABLE-ENTRIES OCCURS 80 TIMES.
           15 RI2-LINE-ENTRY.
                20 RI2-FROM-HWY-AREA.
                     25 RI2-FRM-HWY                          PIC X(16).
                     25 RI2-DIR1                             PIC X(02).
                20 RI2-FROM-HHG-CITY-ST REDEFINES RI2-FROM-HWY-AREA
                                                            PIC X(18).
                20 FILLER                                    PIC X(01).
                20 RI2-MILES                                 PIC ZZZ9.
                20 FILLER                                    PIC X(01).
                20 RI2-TO-LINE.
                     25 FILLER                               PIC X(01).
                     25 RI2-DIR2                             PIC X(02).
                     25 FILLER                               PIC X(01).
                     25 RI2-OF-LIT                           PIC X(02).
                     25 FILLER                               PIC X(01).
                     25 RI2-CITY-ST                          PIC X(21).
                20 RI2-TO-LINE-CITY REDEFINES RI2-TO-LINE.
                     25 FILLER                               PIC X(01).
                     25 RI2-CITY-ST-PROPER                   PIC X(21).
                20 RI2-JCT-LINE REDEFINES RI2-TO-LINE.
                     25 FILLER                               PIC X(01).
                     25 RI2-JCT                              PIC X(03).
                     25 FILLER                               PIC X(01).
                     25 RI2-JCT-INTKEY                       PIC 9(09).
                     25 FILLER                               PIC X(08).
                20 RI2-TO-LINE-LAST REDEFINES RI2-TO-LINE-CITY.
                     25 FILLER                               PIC X(01).
                     25 RI2-CITY-ST-LAST                     PIC X(21).
                20 FILLER                                    PIC X(01).
                20 RI2-TOT-TIME.
                     25 RI2-TOT-HRS                          PIC 9(02).
                     25 RI2-TOT-MIN                          PIC 9(02).
                20 FILLER                                    PIC X(01).
                20 RI2-TOT-MILES                             PIC ZZZZ9.
                20 FILLER                                    PIC X(01).
                20 RI2-COMMENTS.
                     25 RI2-NOTES OCCURS 5 TIMES.
                          30 FILLER                          PIC X(02).
                          30 RI2-NOTE-LAST                   PIC X(01).
```

Route Segment Table - Online

```
02 ROUTE-DISPLAY-SCREEN-LAYOUT.
```

```
   05 TS2-SCREEN-NO                                          PIC 9(02).
   05 TS2-FROM-CITY-ST                                       PIC X(21).
   05 TS2-TO-CITY-ST                                         PIC X(21).
   05 TS2-ROUTE-SEL                                          PIC X(40).
   05 TS2-NUMBER-LINES                                       PIC X(02).
*                      LINE TYPE: D=ROUTE DETAIL
*                                 S=STATE MILEAGE
*                                 G=HHG TRACE RECAP LINE
*                                 H=HIGHLIGHT ROUTE DETAIL LINE
   05 TS2-SCREEN-ENTRIES OCCURS 16 TIMES
      10 TS2-LINE-TYPE                                       PIC X(01).
      10 TS2-LINE-ENTRY                                      PIC X(81).
02 TS2-STORED-END                                           PIC X(01).
@PROC2 =
```

Working Sample Program - Command Level CICS

Refer to member MI9335 on dataset TDM.SRC.T on the system distribution tape for a working sample command level CICS interface program.

Working Sample Program - Batch

Refer to member MI9435 on dataset TDM.SRC.T on system distribution tape for a working sample Batch interface program. Use member MI9435X on dataset TDM.JCL.T to run MI9435.

Working Sample Program-Detail Routes-Command Level CICS

Refer to member MI79335 on dataset TDM.SRC.T on the system distribution tape for a working sample command level CICS interface program.

Working Sample Program-Detail Routes-Batch

Refer to member MI79435 on dataset TDM.SRC.T on the system distribution tape for a working sample batch interface program. Use the member MI79435X on dataset TDM.JCL.T to run MI79435.