

 RAND McNALLY

---

**MileMaker<sup>®</sup> IMS**

# MILEMAKER<sup>®</sup>

**Technical Guide**



# MileMaker<sup>®</sup> IMS

---

# MILEMAKER<sup>®</sup>

All features and functionality described in this document may not be included in the licensed software product. Please refer to your contract for details on the product licensed to you. Distance and other calculations referenced in this document are for illustrative purposes only; they may not reflect actual mileage. Information contained in this document is strictly confidential and proprietary to Rand McNally & Company. For use by authorized licensees only.

Copyright 2008 by Rand McNally & Company, IntelliRoute, and MileMaker are trademarks of Rand McNally & Company. Canadian postal codes © 2008 DMTI Spatial, Inc. All other product names are trademarks of their respective holders.

## Technical Guide

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



---

# Table of Contents

<b>Chapter 1 - Installation Requirements .....</b>	<b>1</b>
<b>General Information .....</b>	<b>2</b>
<b>Operating Environment .....</b>	<b>3</b>
<b>Installation Tapes .....</b>	<b>4</b>
<b>Data Distribution Tapes .....</b>	<b>6</b>
<b>Installation Procedure .....</b>	<b>6</b>
<b>System Gen for Databases (Step 1) .....</b>	<b>7</b>
<b>System Gen for Program and Trans ((7) On-line) ((2) BMP).....</b>	<b>9</b>
<b>Application Gens (Step 2).....</b>	<b>10</b>
<b>DBD Member Names (Required Source Members).....</b>	<b>10</b>
<b>PSB Member Names .....</b>	<b>11</b>
<b>System and Application Gens for Current Users .....</b>	<b>13</b>
<b>Compiles (Step 3).....</b>	<b>13</b>
<b>Recommended Compile Options for IBM Enterprise COBOL.....</b>	<b>13</b>
<b>Compiles for Existing Customers .....</b>	<b>13</b>
<b>On-line Module Compiles with Static Linkages .....</b>	<b>14</b>
<b>Batch Module Compiles with Static Linkages .....</b>	<b>15</b>
<b>Batch Utility Module Compiles .....</b>	<b>16</b>
<b>MFS .....</b>	<b>16</b>
<b>VSAM File Allocation for New Users.....</b>	<b>17</b>
<b>VSAM File Allocation for Existing Users.....</b>	<b>18</b>
<b>Data Base Loads .....</b>	<b>19</b>
<b>Data Base Loads for Current Users .....</b>	<b>22</b>
<b>Installation of IMS MileMaker Guide 19 Library/Data Files .....</b>	<b>28</b>
<b>Creation of IMS MileMaker Libraries/Partitioned Files.....</b>	<b>28</b>
<b>Creation of IMS MileMaker Data Files.....</b>	<b>29</b>
<b>Condensed IMS MileMaker Guide 19 Install Instructions .....</b>	<b>30</b>
<b>Existing Clients Who Wish to Preserve Updates to MileMaker Files.....</b>	<b>30</b>
<b>Existing Clients Who Load All Files from the Distribution Media .....</b>	<b>31</b>
<b>Checklists.....</b>	<b>34</b>
<b>File Data Set Name Worksheet .....</b>	<b>34</b>
<b>Initial Installation Checklist.....</b>	<b>35</b>
<b>Program Description .....</b>	<b>42</b>
<b>Parameter File Records .....</b>	<b>47</b>
<b>Modules / Routines General Information ("revisions") .....</b>	<b>48</b>
<b>Files / Data Bases General Information .....</b>	<b>49</b>
<b>Chapter 2 - Batch Processing .....</b>	<b>51</b>
<b>Batch Mileage/ZIP Inquiry.....</b>	<b>52</b>
<b>Batch System Description .....</b>	<b>52</b>
<b>Record Description: Input Control Record.....</b>	<b>53</b>
<b>Record Description: Format One Input Record .....</b>	<b>54</b>
<b>Record Description: Format Two Input Record .....</b>	<b>55</b>
<b>Record Description: Tape Output Control Record.....</b>	<b>56</b>
<b>Record Description: Tape Output Copyright Records .....</b>	<b>57</b>
<b>Record Description: Tape Output Record (Format One) .....</b>	<b>58</b>

<b>Record Description: Tape Output Record (Format Two)</b> .....	59
<b>Report Output Description</b> .....	60
<b>Error Messages</b> .....	61
<b>Chapter 3 - Interface Documentation</b> .....	63
<b>Mileage/ZIP Interface</b> .....	64
<b>Mileage/ZIP Interface Requirements</b> .....	64
<b>Field Definitions</b> .....	64
<b>Layout of Storage Area (Online)</b> .....	68
<b>Layout of Storage Area (Batch)</b> .....	69
<b>Sample IMS BMP Interface Code Segments</b> .....	70
<b>Chapter 4 - Component File Deletion</b> .....	79
<b>Component File Deletion Program</b> .....	80
<b>Description of YMSMI390 Program</b> .....	80
<b>Sample JCL for YMI390BP</b> .....	81
<b>Input SPLC File Record Format</b> .....	81
<b>Display Examples Produced by YMSMI390 Program</b> .....	82

# INSTALLATION REQUIREMENTS



---

## Chapter Contents

<b>GENERAL INFORMATION</b> .....	2
<b>Operating Environment</b> .....	3
<b>INSTALLATION TAPES</b> .....	4
<b>Data Distribution Tapes</b> .....	6
<b>Installation Procedure</b> .....	6
<b>System Gen for Databases (Step 1)</b> .....	7
<b>System Gen for Program and Trans ((7) On-line) ((2) BMP)</b> .....	9
<b>Application Gens (Step 2)</b> .....	10
<b>DBD Member Names (Required Source Members)</b> .....	10
<b>PSB Member Names</b> .....	11
<b>System and Application Gens for Current Users</b> .....	13
<b>Compiles (Step 3)</b> .....	13
<b>Recommended Compile Options for IBM Enterprise COBOL</b> .....	13
<b>Compiles for Existing Customers</b> .....	13
<b>On-line Module Compiles with Static Linkages</b> .....	14
<b>Batch Module Compiles with Static Linkages</b> .....	15
<b>Batch Utility Module Compiles</b> .....	16
<b>MFS</b> .....	16
<b>VSAM File Allocation for New Users</b> .....	17
<b>VSAM File Allocation for Existing Users</b> .....	18
<b>Data Base Loads</b> .....	19
<b>Data Base Loads for Current Users</b> .....	22
<b>INSTALLATION OF IMS MILEMAKER GUIDE 19 LIBRARY/DATA FILES</b> .....	28
<b>Creation of IMS MileMaker Libraries/Partitioned Files</b> .....	28
<b>Creation of IMS MileMaker Data Files</b> .....	29
<b>CONDENSED IMS MILEMAKER GUIDE 19 INSTALL INSTRUCTIONS</b> .....	30
<b>Existing Clients Who Wish to Preserve Updates to MileMaker Files</b> .....	30
<b>Existing Clients Who Load All Files from the Distribution Media</b> .....	31
<b>CHECKLISTS</b> .....	34
<b>File Data Set Name Worksheet</b> .....	34
<b>Initial Installation Checklist</b> .....	35
<b>Program Description</b> .....	42

---

## **General Information**

This guide will assist in the installation of IMS MileMaker. The release tapes contain the data and modules to process on-line, batch, and BMP mileage requests. The installation requires familiarity in such areas as IBM JCL, VSAM, IMS, and utilities (data base creation, tuning, support). Please call Rand McNally & Company with any question regarding the installation.

### **Section 1**

This section contains installation tape information.

### **Section 2**

This section describes the five installation steps.

### **Section 3**

This section provides some checklists which may be helpful during the installation.

## Operating Environment

The operating environment minimum requirements for IMS MileMaker are as follows :

- IBM z/OS V1.7 operating system
- IMS V8.1
- IBM Enterprise COBOL
- Approximately 650 MB data
- 450 cylinders of 3390 disk storage (estimated) for Guide 19
- 313 cylinders of 3390 disk storage (estimated) for Guide 18
- 763 cylinders of 3390 disk storage (estimated) for support of both guides
- Majority of data bases are 'SHISAM', there are two 'HISAM'

If you have any questions about the installation of this distribution, please contact Rand McNally & Company Customer Service Department at (800) 234-4069.

---

# Installation Tapes

Three tapes will be distributed for an initial installation of the IMS MileMaker system. The cover letter that accompanied the tapes will identify which tape is the system distribution tape and which are the data distribution tapes. Please refer to the cover letter for tape label assignments, density, vol ser numbers, and names.

The system distribution tape contains "PDS" members that were created using a "PANVALET" transfer utility. This separate tape contains the following:

## **"System Distribution Tape" Contents**

The system tape identified by volume CT600 contains the following two files:

1) TDM.SRC.T SOURCE MEMBERS CONTAINING:  
(NOT IN THIS PARTICULAR ORDER)

- A. PROGRAM SOURCE
- B. SOURCE FORMATS (MFS)
- C. ACB SOURCE
- D. PSB SOURCE
- E. DBD SOURCE
- F. SAMPLE COMPILE JCL MEMBERS
- G. SAMPLE JCL MEMBERS
- H. DELETE/DEFINE MEMBER
- I. SYSTEM GEN MEMBERS
- J. INSTALLATION GUIDE

2) TDM.LDL.T DATASET CONTAINING IBM Enterprise COBOL LOAD  
MODULES

Using the IBM utility 'IEBCOPY', copy the PDS's found on your system distribution tape to disk PDS's. These datasets contain all the members for the release.

Sample JCL to load this "system distribution tape" into your system :

```
//TDDDATA JOB ---YOUR JOB CARD
//* COPY DISTRIBUTION MEMBERS TO YOUR SYSTEM
//STEP01 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//JCLIN DD DSN=TDM.SRC.T, TDM SOURCE TAPE
// DISP=(OLD,KEEP,KEEP),
// LABEL=1,
// UNIT=TAPE, - FROM COVER LETTER
// VOL=SER=XXXXXX - FROM COVER LETTER
//JCLOUT DD DSN=(YOUR JCL LIBRARY NAME),
// DISP=(NEW,CATLG,DELETE),
// UNIT=SYSDA, - DASD
// SPACE=(CYL,(15,2,25),RLSE),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=23440,DSORG=PO)
//SYSUT3 DD UNIT=SYSDA,SPACE=(TRK,(1))
//SYSUT4 DD UNIT=SYSDA,SPACE=(TRK,(1))
//SYSIN DD *
COPY INDD=JCLIN,OUTDD=JCLOUT
/*
//STEP02 EXEC PGM=IEBCOPY
//LDLIN DD DSN=TDM.LDL.T, COBOL LOAD MODULES
// DISP=OLD,
// UNIT=TAPE,
// LABEL=(2,SL)
// VOL=(,RETAIN,,,REF=*.STEP01.JCLIN)
//LDLOUT DD DSN=MILEMAKE.LOADLIB,
// DISP=(NEW,CATLG,DELETE),
// UNIT=DISK,
// SPACE=(TRK,(40,,20),RLSE),
// DCB=(RECFM=U,LRECL=0,BLKSIZE=19069,DSORG=PO)
//SYSUT3 DD UNIT=DISK,SPACE=(TRK,(1))
//SYSUT4 DD UNIT=DISK,SPACE=(TRK,(1))
//SYSIN DD *
COPY INDD=LDLIN,OUTDD=LDLOUT
/*
//
```

## Data Distribution Tapes

The "Data Distribution Tape" contains data files for the MileMaker system. These sequential files, created by a utility copy procedure, will be used in data base loads. This tape contains the following data files.

Data TAPE FILE CT601 (36TRK; 3490E CARTRIDGE; SL) -

### Data Distribution Tape

FILE	DSN	RECFM	LRECL	BLKSIZE	CONTENTS
1	TDM.GE010D.T	FB	124	32240	Geographic File
2	TDM.PM010D.T	FB	80	8000	Parameter File
3	TDM.UM010D.T	FB	80	80	User Conversion File
4	TDM.MI390D.T	FB	80	80	Route Component File
5	TDM.MI520D.T	FB	80	80	Billing File
6	TDM.MI020D.T	FB	2036	30540	Locator File
7	TDM.MI350D.T	FB	9	27000	Cross Reference File
8	TDM.MI605D.T	VB	966	32000	Secondary Key Point Mileage File
9	TDM.MI140B.T	FB	4088	28616	Mileage File
10	TDM.MI145B.T	FB	8	32000	Traversal File
11	TDM.MI305D.T	VB	2211	31204	Blocker File
12	TDM.MI340M.T	VB	22	31504	Short Distance Mileage (A-K-B) File
13	TDM.MI015D.T	FB	2000	30000	ZIP Code Matrix File
14	TDM.MI105D.T	FB	31	31000	ZIP Code Master File
15	TDM.MI015D.I	FB	2000	2000	ZIP Code Matrix Init File
16	TDM.MI105D.I	FB	31	31	ZIP Code Master Init File
17	TDM.MI882D.T	FB	60	32700	5-Digit ZIP Reference File

## Installation Procedure

There are five basic steps to install MileMaker:

### 1. System Gens

- DBD
- Program
- Tran
- Data Bases
- Control Region

### 2. Application Gens

- DBD
- PSB
- ACB

3. Compiles And Assembles  
     Programs (optional – a program load library is provided)  
     MFS Formats
4. Delete And Define Data Bases  
     ("IDCAMS" utility for VSAM allocation)
5. Load Databases

## System Gen for Databases (Step 1)

Using your own JCL procedure, generate the DBD for the data bases.

The following is a list of required members:

```

DATABASE DBD=YDUMMY
DATABASE DBD=YGE010D
DATABASE DBD=YGE010X
DATABASE DBD=YGE010Y
DATABASE DBD=YMI015D
DATABASE DBD=YMI020D
DATABASE DBD=YMI022D
DATABASE DBD=YMI105D
DATABASE DBD=YMI140A
DATABASE DBD=YMI140B
DATABASE DBD=YMI145A
DATABASE DBD=YMI145B
DATABASE DBD=YMI305D
DATABASE DBD=YMI307D
DATABASE DBD=YMI340M
DATABASE DBD=YMI342M
DATABASE DBD=YMI350D
DATABASE DBD=YMI352D
DATABASE DBD=YMI390D
DATABASE DBD=YMI520D
DATABASE DBD=YMI605D
DATABASE DBD=YMI607D
DATABASE DBD=YPA010D
DATABASE DBD=YUM010D
DATABASE DBD=YMI882D

```

The following new databases were added for Guide 19:

```

DATABASE DBD=YGE011D
DATABASE DBD=YGE011X
DATABASE DBD=YGE011Y
DATABASE DBD=YMI883D

```

The above members should be created by the client's data base administrator.

---

**Note:** System Gen for databases using your own JCL procedure, generate the control region DDCARDS for the data bases.

---

For the OS control region the same members above must be added to the system. these DDNAMES are associated with their correct 'DBD' names and 'VSAM' cluster names. the 'DBD' names are same as above list and the 'VSAM' names are defined in the 'YDLETDXX' members.

## System Gen for Program and Trans ((7) On-line) ((2) BMP)

Using your own JCL procedure, generate the IMS Gen for the programs and transactions. Note that the following should be revised according to user's teleprocessing considerations and standards.

```
APPLCTN PSB=YGE010,PGMTYPE=(TP,,007)
TRANSACT CODE=YGE010T,PRTY=(01,01,65535), X
MSGTYPE=(MULTSEG,RESPONSE,007),PROCLIM=(00001,00004) X
INQUIRY=(NO,RECOVER),MODE=SNGL,ROUTING=NO

APPLCTN PSB=YPA010,PGMTYPE=(TP,,007)
TRANSACT CODE=YPA010T,PRTY=(01,01,65535), X
MSGTYPE=(MULTSEG,RESPONSE,007),PROCLIM=(00001,00004) X
INQUIRY=(NO,RECOVER),MODE=SNGL,ROUTING=NO

APPLCTN PSB=YUM010,PGMTYPE=(TP,,007)
TRANSACT CODE=YUM010T,PRTY=(01,01,65535), X
MSGTYPE=(MULTSEG,RESPONSE,007),PROCLIM=(00001,00004) X
INQUIRY=(NO,RECOVER),MODE=SNGL,ROUTING=NO

APPLCTN PSB=YMI001,PGMTYPE=(TP,,007)
TRANSACT CODE=YMI001T,PRTY=(01,01,65535), X
MSGTYPE=(MULTSEG,RESPONSE,007),PROCLIM=(00001,00004) X
INQUIRY=(NO,RECOVER),MODE=SNGL,ROUTING=NO

APPLCTN PSB=YMI100,PGMTYPE=(TP,,007)
TRANSACT CODE=YMI100T,PRTY=(01,01,65535), X
MSGTYPE=(MULTSEG,RESPONSE,007),PROCLIM=(00001,00004) X
INQUIRY=(NO,RECOVER),MODE=SNGL,ROUTING=NO

APPLCTN PSB=YMI103,PGMTYPE=(TP,,007)
TRANSACT CODE=YMI103T,PRTY=(01,01,65535), X
MSGTYPE=(MULTSEG,RESPONSE,007),PROCLIM=(00001,00004) X
INQUIRY=(NO,RECOVER),MODE=SNGL,ROUTING=NO

APPLCTN PSB=YMI605,PGMTYPE=(TP,,007)
TRANSACT CODE=YMI605T,PRTY=(01,01,65535), X
MSGTYPE=(MULTSEG,RESPONSE,007),PROCLIM=(00001,00004) X
INQUIRY=(NO,RECOVER),MODE=SNGL,ROUTING=NO

(BMP MEMBERS FOLLOW)
APPLCTN PSB=YMI200BM,PGMTYPE=BATCH
APPLCTN PSB=YMI390BM,PGMTYPE=BATCH
```

The above members should be created by the client's Data Base Administrator.

## Application Gens (Step 2)

### DBD Member Names (Required Source Members)

```
YDUMMY
YGE010D
YGE010X      (SECONDARY INDEX)
YGE010Y      (SECONDARY INDEX)
YGE011D
YGE011X      (SECONDARY INDEX)
YGE011Y      (SECONDARY INDEX)
YMI015D
YMI020D
YMI022D
YMI105D
YMI140A
YMI140B
YMI145A
YMI145B
YMI305D
YMI307D
YMI340M
YMI342M
YMI350D
YMI352D
YMI390D
YMI520D
YMI605D
YMI607D
YPA010D
YUM010D
YMI882D
YMI883D
```

An example of JCL used to generate the DBDs is provided on the release tape (YDBDGEN).

## PSB Member Names

### ON-LINE PROGRAMS

1.	YGE010	GEOGRAPHIC
2.	YPA010	PARAMETER
3.	UM010	USER CONVERSION
4.	YMI001	MASTER MENU
5.	YMI100	MILEAGE
6.	YMI103	ORIGIN
7.	YMI605	SECONDARY

### BATCH LOAD PROGRAMS

8.	YBALOAD1	SECONDARY INDEX INITIALIZATION
9.	YBALOAD2	LOAD OF COMMON DATA BASES
10.	YBALOAD3	LOAD OF LATEST GUIDE DATA BASES
11.	YBALOAD4	INITIALIZATION OF PREVIOUS GUIDE
12.	YBALOAD6	GEOGRAPHIC MERGE
13.	YBALOAD7	SECONDARY EDIT - CURRENT
14.	YBALOAD8	SECONDARY EDIT - PREVIOUS
15.	YBAMRGE7	MERGE CURRENT SECONDARY
16.	YBAMRGE8	MERGE PREVIOUS SECONDARY

### BATCH PROGRAM

17.	YMI200	BATCH MILEMAKER
-----	--------	-----------------

### BMP PROGRAMS

18.	YMI200BM	(SAME AS BATCH (YMI200) EXCEPT FOR "CMPAT=YES")
19.	YMI390BM	DELETE COMPONENT FILE

### NEW BATCH PROGRAMS

20.	YBALOADA	PREVIOUS GUIDE GEO INITIALIZATION
21.	YBALOADB	LOAD OF PREVIOUS GUIDE GEO AND ZIP CODE FILES
22.	YBALOADC	INITIALIZE SECONDARY FILE

An example of JCL used to generate the PSBs is provided on the release tape (YPSBGEN).

Using your own JCL procedure, generate the ACBGEN for the PCBs. The following PCBs for the application's 'ACBLIB' members must be generated.

1.	BUILD PSB=YMI001
2.	BUILD PSB=YMI100
3.	BUILD PSB=YMI103
4.	BUILD PSB=YGE010
5.	BUILD PSB=YPA010
6.	BUILD PSB=YMI605
7.	BUILD PSB=YUM010

8. BUILD PSB=YMI200BM (BMP)
9. BUILD PSB=YMI390BM (BMP)

The above members should be created by the client's Data Base Administrator. an example of JCL used to generate the ACB s is provided on the release tape (YACBGEN)

## System and Application Gens for Current Users

Current users may not need to complete all of the Gen steps. The following may help in deciding which Gens are needed.

```
TRANSACTIONS:  NO CHANGES

PSB:           CHANGES           Changed the following PSBs:
                                           YMI100, YMI103, YMI200,
                                           YMI605
                                           Added the following PSBs:
                                           YBALOADA, YBALOADB,
                                           YBALOADC

DBD:           CHANGES           Changed record sizes for
                                           the following DBDs:
                                           YMI305D, YMI340M, YMI390D,
                                           YDUMMY
                                           Added the following DBDs:
                                           YGE011D, YGE011X, YGE011Y,
                                           YMI883D

ACBGEN:       CHANGES

DATASET
DYNAMIC
ALLOCATION:    CHANGES
```

## Compiles (Step 3)

The MileMaker source and load modules are provided on the release tape. The load modules were created using the IBM Enterprise COBOL compiler under IMS V8.1.

Customers may optionally recompile the COBOL source modules. Note that the load modules do not have the same name as the source modules. For example, source module 'YMSGE010' is compiled and linked to load module 'YGE010'.

Sample compile JCL is provided on the release tape under the member names: YCOBOL1, YCOBOL2, and YCOBOL3.

## Recommended Compile Options for IBM Enterprise COBOL

```
RENT, TRUNC (BIN) , DYNAM
```

## Compiles for Existing Customers

Existing customers may use the COBOL load modules on the release tape, or they may recompile the COBOL source modules on the release tape.

Existing customers must reassemble all MileMaker MFS source members.

## On-line Module Compiles with Static Linkages

Compile and link the following modules in the sequence provided. Be sure to include the proper modules in each link.

(C) = "CBLTDLI"; (D) = "DLITCBL" REPRESENT REQUIRED INCLUDE STATEMENT

	SOURCE NAME	OBJECT NAME	COMPILE SEQ
1.	YMSGE010 (C,D) INCLUDE (YMIABBRV)	"YGE010"	#11
2.	YMSMIABB	"YMIABBRV"	#1
3.	YMSMI001 (C,D)	"YMI001"	#10
4.	YMSMI100 (C,D) INCLUDE (YMI105) INCLUDE (YMI115) YMI116 INCLUDE (YMIABBRV) INCLUDE (YMI120) YMI121 INCLUDE (YMI125) YMI126 INCLUDE (YMI140) YMI141	"YMI100"	#9
5.	YMSMI103 (C,D) INCLUDE (YMI105) INCLUDE (YMI115) YMI116 INCLUDE (YMIABBRV) INCLUDE (YMI120) YMI121 INCLUDE (YMI125) YMI126 INCLUDE (YMI140) YMI141	"YMI103"	#8
6.	YMSMI105 (C)	"YMI105"	#6
7.	YMSMI110 (C) INCLUDE (YMIABBRV) INCLUDE (YMI105) INCLUDE (YMI115) YMI116 INCLUDE (YMI120) YMI121 INCLUDE (YMI125) YMI126 INCLUDE (YMI140) YMI141 (INTERFACE MODULE, LINKED WITH USERS MODULE)	"YMI110"	#7
8.	YMSMI115 (C) & YMSMI116	"YMI115" & YMI116	#5
9.	YMSMI120 (C) & YMSMI121	"YMI120" & YMI121	#4
10.	YMSMI125 (C) & YMSMI126	"YMI125" & YMI126	#3
11.	YMSMI140 (C) & YMSMI141	"YMI140" & YMI141	#2
12.	YMSMI605 (C,D) INCLUDE (YMIABBRV)	"YMI605"	#12

	SOURCE NAME	OBJECT NAME	COMPILE SEQ
13.	YMSPA010 (C,D)	"YPA010"	#13
14.	YMSUM010 (C,D) INCLUDE (YMIABBRV)	"YUM010"	#14

## Batch Module Compiles with Static Linkages

	SOURCE NAME	OBJECT NAME	COMPILE SEQ
1.	YMSMI200 (C,D) * INCLUDE (YMI201) YMI202 * INCLUDE (YMI205) * INCLUDE (YMI215) YMI216 * INCLUDE (YMI220) YMI221 * INCLUDE (YMI225) YMI226 * INCLUDE (YMI230) YMI231 * INCLUDE (YMI240) YMI241	"YMI200"	#9
2.	YMSMI201 (C) & YMSMI202	"YMI201" & YMI202	#1
3.	YMSMI205 (C)	"YMI205"	#7
4.	YMSMI210 (C) * INCLUDE (YMI201) YMI202 * INCLUDE (YMI205) * INCLUDE (YMI215) YMI216 * INCLUDE (YMI220) YMI221 * INCLUDE (YMI225) YMI226 * INCLUDE (YMI240) (INTERFACE MODULE, LINKED WITH USERS MODULE) YMI241	"YMI210"	#8
5.	YMSMI215 (C) & YMSMI216	"YMI215" & YMI216	#6
6.	YMSMI220 (C) & YMSMI221	"YMI220" & YMI221	#5
7.	YMSMI225 (C) & YMSMI226	"YMI225" & YMI226	#4
8.	YMSMI240 (C) & YMSMI241	"YMI240" & YMI241	#2
9.	YMSMI230 (C) & YMSMI231	"YMI230" & YMI231	#3
	<b>* BMP MODULE(S)</b>		
1.	YMSMI20B (C,D) (WITH BMP CHGS. FOR LINKAGE)	"YMI200BM"	
2.	YMSMI390 (C,D) (DELETE COMPONENT FILE BMP PGM)	"YMI390BM"	

## Batch Utility Module Compiles

	SOURCE NAME	OBJECT NAME
1.	YRANLD1 (C,D)	"YRANLD1"
2.	YRANLD2 (C,D)	"YRANLD2"
3.	YRANLD3 (C,D)	"YRANLD3"
4.	YRANLD4 (C,D)	"YRANLD4"
5.	YRANLD6 (C,D)	"YRANLD6"
6.	YRANLD7 (C,D)	"YRANLD7"
7.	YRANLD8 (C,D)	"YRANLD8"
8.	YMERGE7 (C,D)	"YMERGE7"
9.	YMERGE8 (C,D)	"YMERGE8"
10.	YRANLDA (C,D)	"YRANLDA"
11.	YRANLDB (C,D)	"YRANLDB"
12.	YRANLDC (C,D)	"YRANLDC"

## MFS

MFS format assemble procs are dependent on the environment, so they are not provided. Assemble member "YGE10MFS" to "GE10DF"(DIF) "GE10MI"(MID) "GE10"(MOD) for example.

1.	YGE10MFS
2.	YGE11MFS
3.	YMI01MFS
4.	YMI10MFS
5.	YMI13MFS
6.	YMI65MFS
7.	YPA10MFS
8.	YPA12MFS
9.	YPA13MFS
10.	YPA15MFS
11.	YPA22MFS
12.	YUM10MFS
13.	YUM11MFS
14.	YMI11MFS

MFS MEMBERS	TRANSACTION(S)	COMMENTS
GE010	'YGE010T'	GEOGRAPHIC FILE UPDATE
GE011	'YGE010T'	GEOGRAPHIC FILE INQUIRY
MI001	'YMI001T'	MASTER MENU SCREEN
MI100	'YMI100T'	MILEAGE INQUIRY
MI103	'YMI103T'	ORIGIN INQUIRY
MI605	'YMI605T'	SECONDARY KEYPOINT
PA010	'YPA010T'	PARAMETER SUB-MENU
PA012	'YPA010T'	PARAMETER (PASSWORD)
PA013	'YPA010T'	PARAMETER (USER PROFILE)
PA015	'YPA010T'	PARAMETER (DEFINITIONS)
PA022	'YPA010T'	PARAMETER (TERMINALS)

<b>MFS MEMBERS</b>	<b>TRANSACTION(S)</b>	<b>COMMENTS</b>
UM010	'YUM010T'	USER CONVERSION (UPDATE)
UM011	'YUM010T'	USER CONVERSION (BROWSE)

## **VSAM File Allocation for New Users**

Sample allocation JCL is provided on the release tape. This JCL, after review and modification, may be used to allocate MileMaker files. although the previous guide may not be supported, all MileMaker files must be allocated and available to the system.

New MileMaker clients will follow the steps outlined below. Existing users have additional considerations. These are explained after step 'E'. We recommend existing users read page 17 for familiarization.

Clients will run either step 'A' or step 'B'. If both guides are supported, then run step 'A'. If only the current guide is supported, then run step 'B'. Do not execute both steps.

### **Step A**

Function : VSAM file allocation for users intending to use both Previous and Current Guides.

JCL : YDLETDEF

### **Step B**

Function : VSAM file allocation for users intending to use only the current guide.

JCL : YDLETDF

### **Step C**

Function : VSAM file allocation for the dummy, parameter, user conversion, and billing files.

JCL : YDLETDF

### **Step D**

Function : VSAM file allocations for files necessary to process 3-digit ZIP Code based mileages. If ZIP Code mileages are not needed, the 'records' parameter should be changed from 950 to 1. Do this for both ZIP files.

JCL : YDLETDF

### **Step E**

Function : VSAM file allocations for the Geographic, 5-digit ZIP, and Secondary files. If the previous guide is not supported, do not allocate the previous secondary file. This is accomplished by removing the 'delete/define' statements for the file named 'TDMV.YMI607DK'.

JCL : YDLETDF

### **Step F**

Function : VSAM file allocations for the previous Guide Geographic, and ZIP Code files.

JCL : YDLETDIG

### **Step G**

Function : VSAM file allocation for the YDUMMY file.

JCL : YDLETDIF

## **VSAM File Allocation for Existing Users**

Current users have several options to consider when allocating files. Please read the following explanations before proceeding.

### **Step A / Step B**

Users must run either step 'A' or step 'B' as outlined earlier. This depends on whether or not the previous guide is supported. These files will be loaded with new data reflecting the current mileages.

### **Step C**

Since the files allocated in this step have not changed, users may skip this step. If this step is skipped, then check the common file load (step 'B'; data base loads) to ensure that these files are not loaded. This is done by removing the control cards for the four files (YPA010D, YUM010D, YMI520D, YDUMMY). Users may run this step if desired, but existing parameter security records and any user conversion records would be lost.

### **Step D**

Users will run this step as outlined earlier.

### **Step E**

This step allocates the Geographic, 5-digit ZIP, and Secondary files. Users need to decide if any previous updates done to these files need to be saved. If no updates have occurred, or if the user updates to these files are not needed, this step may be run as outlined earlier. If updates are needed, please read the following:

#### **Geographic File**

In order to save changes, a sequential copy of the user's geographic data base is needed (see data base loads for current users; Step 'B'; Geographic File). Be sure this sequential copy is available before deleting any geographic data bases. After the copy is made, the Geographic File allocation may be run.

#### **Secondary - Current**

In order to save changes, the current secondary data base is updated and merged with our file (see data base loads for current users; Step D; secondary data base). If users wish to save file modifications, then do not allocate this secondary file. This is accomplished by removing the 'DELETE/DEFINE' statements for the file named 'TDMV.YMI605DK'.

### **Secondary - Previous**

In order to save changes, the previous secondary data base is loaded with the user's data from the current secondary data base (see data base loads for current users; Step E; previous secondary data base). In all cases the previous secondary file will be loaded, so the allocation should be run as outlined earlier.

### **Step F**

Run this step to create the previous Guide Geographic (YGE011D) and ZIP Code (YMI883D) files.

### **Step G**

Run this step to create the YDUMMY file with the new record size.

## **Data Base Loads**

The data, load programs, and sample JCL are provided on the release tape. Users are responsible for loading the data bases. The load programs will display record counts which may be used to help verify successful loads.

Two Guides (17 and 18) may be supported by the user. If the Previous Guide will not be used, certain files need only be initialized. This will save dasd space. In either case all data bases must be present to the system whether loaded or only initialized. The JCL provided may not be suitable for all installations. Modifications may be necessary to certain areas such as load libraries, buffer space, control cards, and data set names.

New MileMaker clients and existing clients, who have NOT applied user updates, will follow steps 'A' thru 'D' as written.

Existing clients, who have applied user updates, have additional considerations. These are outlined starting on page 22 and should be carefully read.

### **Step A**

Function: Geographic File Secondary Index Initialization  
JCL: YDLIBTC1  
Module: YRANLD1  
PSB: YBALOAD1

After making necessary JCL changes, run this job and verify correct processing. The displayed message will indicate a record count of one.

### **Step B**

Function: Load Of Common Files  
JCL: YDLIBTC2  
Module: YRANLD2  
PSB: YBALOAD2

This job will load the eight files common to both guides. The data bases may be loaded in one run or may be loaded one at a time. This is controlled by the 'CARDIN DD \*' input found in the JCL. The module will load the data bases whose names are found in the 'CARDIN DD \*' input stream. After making any necessary changes to the JCL, run this job and verify correct processing. Four files (YMI390D, YMI520D, YDUMMY, YUM010D) Will be initialized to one record each. The remaining four files (YGE010D, YPA010D, YMI015D, YMI105D) will be initialized to the record counts contained on the cover letter.

The ZIP3 Master, ZIP3 Matrix, and ZIP5 Cross-reference files loaded in this step are necessary to process ZIP3 and ZIP5 code-based mileages. If a user does not need ZIP-based mileages, these files may be initialized to one record each. This will save dasd. To initialize ZIP3 Matrix and Master files, the load JCL (YDLIBTC2) must be modified. In JCL member 'YDLIBTC2' substitute the ZIP3 initialization data sets for the tape data sets (see comments in JCL). Loading the geographic data base may take several hours (wall time) due to the number of records and the secondary indices. The following data bases are loaded in this step:

YGE010D	GEOGRAPHIC
YPA010D	PARAMETER
YUM010D	USER CONVERSION
YMI390D	ROUTE
YMI520D	BILLING
YDUMMY	DUMMY
YMI015D	ZIP MATRIX
YMI105D	ZIP MASTER
YMI882D	ZIP5 CROSS-REFERENCE

### Step C

Function:	Previous Guide (18) File Copies or File Initializations
JCL:	YPRV180: Copies current production files to previous Guide files using IDCAMS REPRO. YDLIBTC4: Initializes the previous Guide files.
Modules:	IDCAM REPRO Utility YRANLD4 (PSB: YBALOAD4)

If previous Guide will be supported, then run the job YPRV180 which copies your seven production Guide 18 files to the Guide 19 previous Guide files.

If previous Guide will not be supported, then your Guide 19 previous Guide files need to be initialized by running the job YDLIBTC4 using "INIT" as the parameter card after the statement: CARDIN DD \*.

YMI352D	CROSS REFERENCE
YMI307D	BLOCKER
YMI022D	LOCATOR
YMI140A	MILEAGE
YMI145A	TRAVERSAL
YMI607D	SECONDARY
YMI342M	SHORT DISTANCE

### Step D

Function: Guide 18 Data Base Loads (Current)  
JCL: YDLIBTC3  
Module: YRANLD3  
PSB: YBALOAD3

This step will load the seven files necessary to process Guide18. All seven files are required and must be loaded. the data bases may be loaded in one run or one at a time. This is controlled by the 'CARDIN DD \*' input stream. After making any necessary JCL changes, run this job and verify correct processing. The correct record counts are found in the cover letter. The following data bases are loaded in this step:

YMI350D	CROSS REFERENCE
YMI305D	BLOCKER
YMI020D	LOCATOR
YMI140B	MILEAGE
YMI145B	TRAVERSAL
YMI605D	SECONDARY
YMI340M	SHORT DISTANCE

### Step E

Function: Initialize previous Guide Geographic indexes  
JCL: YDLIBTCA  
Module: YRANLDA  
PSB: YBALOADA

### Step F

Function: Load previous Guide Geographic and Zip Code files  
JCL: YDLIBTCB  
Module: YRANLDB  
PSB: YBALOADB

### Step G

Function: Initialize YMI605D file  
JCL: YDLIBTCC  
Module: YRANLDC  
PSB: YBALOADC

## Data Base Loads for Current Users

Existing users have some file load issues to consider which new users do not. These involve updating files with new release data while preserving any user updates that may have occurred. If no user updates have occurred, or if these updates are no longer needed, then file loading may proceed as outlined in the previous four steps. Loading files in this manner will, of course, eliminate any client updates previously done. This may be appropriate for many users. if, however, past client updates are

wanted, then some modifications are necessary to the preceding four steps. These load modifications are explained below.

### **Step A: Geographic File Secondary Index Initialization**

If users wish to save file changes, do not run this step at this time (see step 'B'; load of common files; Geographic File) otherwise, this may be run as outlined.

### **Step B: Load Of Common Files**

#### **YMI520D - Billing File**

No changes to this file are contained in the release. Users may initialize this file to one record as explained in the common file load section, or may simply leave the file untouched. If the file is to be initialized, then be sure to run the appropriate IDCAMS DELETE/DEFINE.

### **YMI390D - Route File**

The Route File, if used, stores mileages. With the new release these mileages may be incorrect. This file must be initialized during the load to delete any incorrect mileages. Run the appropriate IDCAMS DELETE/DEFINE prior to the load job.

### **YUM010D - User Conversion**

No changes to this file are contained in the release. Clients may have updated this file so to preserve these changes the file should not be initialized. This file may be left untouched. If the file is to be initialized, then be sure to run the appropriate IDCAMS DELETE/DEFINE.

### **YPA010D**

If client updates were made, then this file must be left alone and must NOT be loaded from the distribution tape.

### **YMI882D**

Changes were made to this file in this release. The file must be loaded from the distribution tape.

### **YGE010D - Geographic File**

The geographic file, like the parameter file, may contain client updates. If users wish to save their file changes, a file merge must be run prior to the load program. The sequential geographic file provided by Rand McNally & Company is merged with a sequential copy of the client geographic file. This new file created by the merge is used as input to the load program. The steps to do this are outlined below.

- A) Create a sequential copy of your geographic data base in primary key order. This may be done using the following.  
JCL: YDLIBTC5  
MODULE: IDCAMS
- B) Run the provided merge program.  
JCL: YDLIBTC6  
MODULE: YRANLD6  
PSB: YBALOAD6
- C) Run the Geographic Secondary Index Initialization, as instructed on page 19, by running YDLIBTC1.
- D) Substitute the sequential file created in 'B' (above) into the load JCL (YDLIBTC2). Run the geographic file load as instructed on page 19, (data Base Loads).

---

**Attention:**

The merge program (YRANLD6) may update the User Conversion (YUM010D) File. A backup of this file is appropriate.

---

## Geographic File Merge Description

The YDLIBTC6 procedure executes the YRANLD6 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
YGE010D	DATA BASE	Guide 17 Geographic File
GE010T	SEQ	Guide 18 Geographic File from Rand McNally Data Distribution Tape.

Output:

DDNAME	TYPE	DESCRIPTION
GE010N	SEQ	Merged Geographic File for use in Guide 18. This file is used as input to the GE010DVD procedure.
YUM010D	DATA BASE	Guide 18 User Conversion File.

This procedure merges the Rand McNally & Company distributed Guide 18 GE010D file with a copy of the user's YGE010D, created by the procedure YDLIBTC5

Six reports are printed by the YRANLD6 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 YGE010D 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 (YUM010D) 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 (YUM010D) 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 YUM010D 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 YUM010D 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 YUM010D 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
YUM010D city records added	YUM010D city records changed
YUM010D SPLC records added	YUM010D 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 distribution records read =
  - Unmatched Rand McNally distribution records
  - + Matched geographic records
  
3. Geographic records written =
  - Unmatched Rand McNally distribution records
  - + Unmatched customer geo records
  - + Matched geographic records
  - Matching internal keys dropped
  - City name changes

### **Step C : Previous Guide (17) Data Base Loads**

The Previous Secondary File (YMI607D) may contain user updates. If users wish to save these changes, the following steps apply.

- A) Allocate Previous Secondary File. (This was done in VSAM FILE allocation; Step 'E', on page 17.)
- B) Copy the existing Current Secondary (YMI605D) into the newly allocated Previous Secondary (YMI607D). It is important to copy the current secondary before any release changes are applied. Users may simply copy the existing current secondary from the production environment.
- C) Run the following job.
  - FUNCTION: In-place update and merge of user's Previous Secondary file.
  - JCL: YDLIBTC8
  - MODULES: YRANLD8 - PREVIOUS SECONDARY FILE  
UPDATE
  - PSB: YBALOAD8

#### **Attention:**

The geographic data base must be current with the release before this job is run. This job will update both the Secondary and Geographic data bases, so a backup of these files is appropriate. Users may want to re-org the Secondary File after this job.

All other data bases in step 'C', except YMI607D, (page 21) will be loaded as instructed.

### **Step D : Current Guide (18) Data Base Loads**

FUNCTION: Initialize Secondary Keypoint File  
YMI605D.  
JCL: YDLIBTCC  
MODULES: YRANLDC - SECONDARY FILE Initialization  
PSB: YBALOADC

### **Step E : Initialize previous Guide Geographic indexes**

JCL: YDLIBTCA  
Module: YRANLDA  
PSB: YBALOADA

### **Step F : Load previous Guide Geographic and Zip Code files**

JCL: YDLIBTCB  
Module: YRANLDB  
PSB: YBALOADB

---

## **Installation of IMS MileMaker Guide 19 Library/Data Files**

*Please read these supplemental instructions for loading the MileMaker Guide 18 application installation from CD-ROM. Once the initial load is complete, you may consult the IMS 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 IMS MileMaker 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 source library called MM19.IMS.SRCLIB.XMIT and a load library called MM19.IMS.LDLLIB.XMIT.

- 1.** Copy the XMIT 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 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 IMS MileMaker Data Files

The MileMaker fixed and variable length files on the CD-ROM are in XMIT format and identified as MM19.IMS.xxxxxx.XMIT where xxxxxx is the base file name. There are 17 MileMaker IMS data files.

1. Copy the MileMaker data XMIT files to your PC's hard drive.
2. Before uploading set the mainframe file attributes of each data file to the following: LRECL=80, RECFM=FB, BLKSIZE=80, 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 disk data files. Refer below to the sample TSO RECEIVE JCL.
5. Follow the installation steps in the Technical Guide to create the MileMaker IMS/VSAM files from the disk data files created by TSO RECEIVE.

### IMS Sample Batch TSO RECEIVE JCL

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

/*

//
```

1.

---

## Condensed IMS MileMaker Guide 19 Install Instructions

### Existing Clients Who Wish to Preserve Updates to MileMaker Files

1. Run either job **YDLETDEF** or job **YDLETDF** in order to define the Guide 19 current and previous Guide mileage files. **YDLETDF** defines dummy previous Guide mileage files when the previous Guide is not used.  
The Guide 19 mileage files are YMI020D, YMI140B, YMI145B, YMI305D, YMI340M, YMI350D, and YMI390D.  
The previous Guide 18 mileage files are YMI022D, YMI140A, YMI145A, YMI307D, YMI342M and YMI352D.  
The record sizes for the following files were changed for Guide 19: YMI305D, YMI340M, and MI390D.
2. Run the job **YDLETDIG** in order to define the following new files for Guide 19:  
YGE011D - previous guide geographic file  
YGE011DX - previous guide geographic secondary index  
YGE011DY - previous guide geographic secondary index  
YMI883D - previous guide ZIP Code file
3. Run the job **YDLETDJG** in order to define the file YDUMMY. Its record size changed for Guide 19.
4. No need to run the job **YDLETDF**, which defines the files, YMI520D, YPA010D, and YUM010D. These files haven't changed for this release. You may use your existing files.  
However, if the billing file, YMI520D, is being used, then it should be reinitialized for this release.  
YPA010D contains user specific security records.  
YUM010D contains user conversion records.
5. No need to run the job **YDLETDF** in order to define the Zip3 mileage files, YMI015D and YMI105D. These files haven't changed for this release.
6. Run the job **YDLETDF** to define the Guide 19 files YGE010D, YMI882D, YMI605D and YMI607D.  
YMI882D is the 5-digit ZIP Code file.  
YGE010D is the geographic file.  
YMI605D and YMI607D are the current and previous secondary mileage files.
7. Run either the job **YPRV180** or the job **YDLIBTC4**.  
**YPRV180** copies the production Guide 18 mileage files to the Guide 19 previous Guide mileage files.

**YDLIBTC4** initializes the Guide 19 previous guide files with assumption that the previous Guide will not be used.

- 8.** Run the job **YDLIBTC3** to load the Guide 19 Mileage files from the distribution media.  
The only files that should be loaded are YMI350D, YMI305D, YMI020D, YMI140B, YMI145B and YMI340M. Note that YMI605D is not loaded at this time.
- 9.** Run the job **YDLIBTC5** to create a sequential copy of your existing production Guide 18 geographic file, YGE010D, containing user updates and your existing Guide 18 ZIP Code file, YM1882D. The sequential geographic file will be input to the jobs **YDLIBTC6** and **YDLIBTCB**. The sequential ZIP Code file will be input to the job **YDLIBTCB**.
- 10.** Run the job **YDLIBTC6** to merge the geographic file from job **YDLIBTC5** with the Rand McNally G19 geographic file. The resulting output geographic sequential file becomes input to the job **YDLIBTC2**.  
The MileMaker file, YUM010D, is updated by **YDLIBTC6**.
- 11.** Run the job **YDLIBTC1** to initialize the secondary indexes of the Guide 19 geographic file YGE010D.
- 12.** Run the job **YDLIBTC2** to load only the following MileMaker Guide 19 files:  
YMI390D - Input file comes from the distribution file.  
YGE010D - Input file comes from job **YDLIBTC6**.  
YMI882D - Input file comes from the distribution file.  
YMI520D - Optional file comes from the distribution file.  
YDUMMY – MileMaker work file.
- 13.** Run the job **YDLIBTCA** to initialize the secondary indexes of the new previous guide geographic file YGE011D.
- 14.** Run the job **YDLIBTCB** to load the following new previous guide files:  
YGE011D - Input sequential file comes from job **YDLIBTC5**.  
YMI883D -Input sequential file comes from job **YDLIBTC5**.
- 15.** Run the job **YDLIBTCC** to initialize the Guide 19 secondary file YMI605D.  
There will be no Secondary file records for Guide 19.

## **Existing Clients Who Load All Files from the Distribution Media**

- 1.** Run either job **YDLETDEF** or job **YDLETDFD** in order to define the Guide 19 current and previous Guide mileage files. **YDLETDFD** defines dummy previous Guide mileage files when the previous Guide is not used.  
The Guide 19 mileage files are YMI020D, YMI140B, YMI145B, YMI305D, YMI340M, YMI350D, and YMI390D.  
The previous Guide mileage files are YMI022D, YMI140A, YMI145A, YMI307D, YMI342M and YMI352D.  
The record sizes for the following files were changed for Guide 19: YMI305D, YMI340M, and MI390D.

2. Run the job **YDLETDIG** in order to define the following new files for Guide 19:  
 YGE011D - previous guide geographic file  
 YGE011DX - previous guide geographic secondary index  
 YGE011DY - previous guide geographic secondary index  
 YMI883D - previous guide ZIP Code file
3. Run the job **YDLETDJG** in order to define the file YDUMMY. Its record size changed for Guide 19.
4. No need to run the job **YDLETDGF**, which defines the files, YMI520D, YPA010D, and YUM010D. These files haven't changed for this release. You may use your existing files.  
 However, if the billing file, YMI520D, is being used, then it should be reinitialized for this release.  
 YPA010D contains user specific security records.  
 YUM010D contains user conversion records.
5. No need to run the job **YDLETDHF** in order to define the Zip3 mileage files, YMI015D and YMI105D. These files haven't changed for this release.
6. Run the job **YDLETDIF** to define the Guide 19 files YGE010D, YMI882D, YMI605D and YMI607D.  
 YMI882D is the 5-digit ZIP Code file.  
 YGE010D is the geographic file.  
 YMI605D and YMI607D are the current and previous secondary mileage files.
7. Run either the job **YPRV180** or the job **YDLIBTC4**.  
**YPRV180** copies the production Guide 18 mileage files to the Guide 19 previous Guide mileage files.  
**YDLIBTC4** initializes the Guide 19 previous guide files with assumption that the previous Guide will not be used.
8. Run the job **YDLIBTC3** to load the Guide 19 Mileage files from the distribution media.  
 The only files that should be loaded are YMI350D, YMI305D, YMI020D, YMI140B, YMI145B, YMI340M and YMI605D.
9. Run the job **YDLIBTC5** to create a sequential copy of your existing production Guide 18 geographic file, YGE010D, containing user updates and your existing Guide 18 ZIP Code file, YMI882D. The sequential geographic file and ZIP Code file will be input to the job **YDLIBTCB**.
10. Run the job **YDLIBTC1** to initialize the secondary indexes of the Guide 19 geographic file YGE010D.
11. Run the job **YDLIBTC2** to load only the following MileMaker Guide 19 files:  
 YMI390D - Input file comes from the distribution file.  
 YGE010D - Input file comes from the distribution file.  
 YMI882D - Input file comes from the distribution file.  
 YMI520D - Optional file comes from the distribution file.  
 YDUMMY – MileMaker work file.

- 12.** Run the job **YDLIBTCA** to initialize the secondary indexes of the new previous guide geographic file YGE011D.
- 13.** Run the job **YDLIBTCB** to load the following new previous guide files:  
YGE011D - Input sequential file comes from job **YDLIBTC5**.  
YMI883D -Input sequential file comes from job **YDLIBTC5**.
- 14.** Run the job **YDLIBTCC** to initialize the Guide 19 secondary file YMI605D.  
There will be no Secondary file records for Guide 19.

---

# Checklists

## File Data Set Name Worksheet

DSN	YOUR DATASET NAME	CONTENTS
	COMMON FILES	
TDMV.YGE010DK		GEOGRAPHIC FILE
TDMV.XGE010DS		SECONDARY INDEX (GEOGRAPHIC)
TDMV.YGE010DS		SECONDARY INDEX (GEOGRAPHIC)
TDMV.YPA010DK		PARAMETER FILE
TDMV.YUM010DK		USER CONVERSION
TDMV.YMI390DK		ROUTE COMPONENT FILE
TDMV.YMI520DK		BILLING FILE
TDMV.YDUMMY		DUMMY/WORK FILE (ONLINE ONLY)
TDMV.YMI015DK		ZIP MATRIX FILE
TDMV.YMI105DK		ZIP MASTER FILE
TDMV.YMI882D		ZIP5 CROSS REFERENCE
	GUIDE 19 FILES	
TDMV.MI020D.T		LOCATOR FILE
TDMV.MI140B.T		MILEAGE FILE
TDMV.MI145B.T		TRAVERSAL FILE
TDMV.MI305D.T		BLOCKER FILE
TDMV.MI340M.T		SHORT DISTANCE
TDMV.MI350D.T		CROSS REFERENCE FILE
TDMV.MI605D.T		SECONDARY KYPT
	GUIDE 18 FILES	
TDMV.MI022D.T		LOCATOR FILE
TDMV.MI140A.T		MILEAGE FILE
TDMV.MI145A.T		TRAVERSAL FILE
TDMV.MI307D.T		BLOCKER FILE
TDMV.MI342M.T		SHORT DISTANCE
TDMV.MI352D.T		CROSS REFERENCE FILE
TDMV.MI607D.T		SECONDARY KYPT
TDMV.YGE011DK		PREVIOUS GUIDE GEOGRAPHIC FILE
TDMV.XGE011DS		PREVIOUS GUIDE SEC INDEX 1
TDMV.YGE011DS		PREVIOUS GUIDE SEC INDEX 2
TDMV.YMI883D		PREVIOUS GUIDE ZIP FILE

## Initial Installation Checklist

- Step 1**            **Review All Documentation**
  
- Step 2A**            **Unload "TDM.SRC.T" from the Distribution Tape to your Library PDS Members.**
  - Check Condition Code of Step Above.
  - Verify Number of Members Versus Cover Letter Members.
  - Check and Change All Members for Naming Convention and User-related Standards and Procedures
  
- Step 2B**            **Unload "TDM.LDL.T" from the Distribution Tape to your load PDS.**
  
- Step 3**            **Request Systems Gens (Macro) from your DBA Group.**
  - Data Bases (List shown in Section 2, Step 1.)
  - Programs (List shown in Section 2, Step 1.)
  - Transactions (List shown in Section 2, Step 1.)
  - Data Base Gen for Control Region or Dynamic Allocation if used. (List shown in Section 2, Step 1.)
  - Verify Condition Codes for All Members.
  - Verify Correct Libraries for Members.
  
- Step 4**            **Set Up Application Gens for Processing**
  - Process DBD Gen.(List shown in Section 2, Step 2.)
  - Process PSB Gen.(List shown in Section 2, Step 2.)
  - Process ACB Gen.(List shown in Section 2, Step 2.)
  - Verify Condition Codes for All Members.
  - Verify Correct Libraries for Members.
  
- Step 5**            **(Optional) Compile MileMaker COBOL source**
  - Compile On-line Related Programs (List shown in Section 2, Step 3.) Watch Sequence of Compile Members for Linkage Considerations (unresolved members).
  - Verify Return Codes on Compile/Link Steps.

- Verify Members Linked into Correct Load Library.
- Compile Batch-related Programs. (List shown in Section 2, Step 3.) Watch Sequence of Compile Members for Linkage Considerations (unresolved members).
- Verify Return Codes on Compile/Link Steps.
- Verify Members Linked into Correct Load Library.
- Compile BMP-related Program (List shown in Section 2, Step 3.) Note the Changes Made to Batch Source Members - Refer to Step 10 of this Checklist.
- Verify Return Codes on Compile/Link Steps.
- Verify Members Linked into Correct Load Library
- Compile Utility Programs (List shown in Section 2, Step 3.)
- Verify Return Codes on Compile/Link Steps.
- Verify Members Linked into Correct Load Library
  
- Step 6      Assemble MFS Members**
- (List shown in Section 2, Step 3.)
- Verify Condition Codes on Mapping Members into Correct MFS Library.
  
- Step 7      Delete/Define for Data Bases**
- (List shown in Section 2, Step 4.)
- Verify Condition Codes and Proper Use of Volumes or DASD in Setting up Space Requirements According to DBA Considerations (recommended VSAM points) from the DBD Gen.
- Verify Cluster and Index Names.
  
- Step 8      Process Initialization of Secondary Indices for Data Bases (Geographic).**
- Set up Load JCL Step (YDLIBTC1)- Secondary Index File(s).
- Verify Correct Control Card for Data Base Name(s) that are to be Initialized at this Time.
- Submit Job for processing.
- Verify Condition Codes and/or Display Messages.
- Unload "System Data Tapes".

- Step 9            Process Common Data Base Loads**
  - Set up Load JCL Step (YDLIBTC2)- Common Files.
  - Verify Correct Control Card for Data Base Name(s) that are to be Initialized/Loaded at this Time.
  - Submit Job for processing.
  - Verify Condition Codes and/or Display Messages.
  
  - Step 10            Process Previous Data Base Copies**
  - Set up Load JCL (YDLIBTC4 Member) – YPRV18D Previous Guide.
  - Verify Correct Control Card for Data Base Name(s) that are to be Loaded at this Time.
  - Submit Job for processing. Note that there will be Long Running, Large Volume Activity
  - It is Recommended that Processing be Split up into Two Separate Job Submits.
  - Verify Condition Codes and/or Display Messages for any Error Conditions.
  - Verify Record Counts from Load Program to Count Given on Cover Letter for Each File.
  
  - Step 11            Process Current Data Base Loads**
  - Set up Load JCL (YDLIBTC3 Member) - Current Guide.
  - Verify Correct Control Card for Data Base Name(s) that are to be Loaded at this Time.
  - Submit Job for processing. Note that there will be Long Running, Large Volume Activity
  - It is Recommended that Processing be Split up into Two Separate Job Submits.
  - Verify Condition Codes and/or Display Messages for any Error Conditions.
  - Verify Record Counts from Load Program to Count Given on Cover Letter for Each File.
  
  - Step 12            Batch Message Processing**
  - Batch Source Revised - Created BMP Member and JCL.
- Check the Following for YMI200 BMP Processing:
- A)    IMS Gen for the BMP-Transaction in IMS Control Region.
  - B)    ACB Gen for the BMP-Transaction in IMS Control Region.

- C) BMP PSB Member (YMI200BM) to be Gen'd.
- D) Assign Certain Class for BMP/IMS Region.
- E) BMP JCL Member (YMI200BP) to be used. (Change Class for BMP Processing.)
- The Batch Source Program has Changed as Follows:
- F) Batch Program (YMSMI200) Revised to BMP Considerations. BMP Program (YMSMI20B) is the Source Member.
  - I) Compile the Revised Source (above) with Required Include Linkage Members. (same as Batch)
  - II) Compile Program to Unique Name "YMI200BM" which is Referenced within BMP JCL
- G) Review JCL Member for Correct Libraries and Parameters.

Check the Following for YMI390 BMP Processing:

- A) IMS Gen for the BMP-Transaction in IMS Control Region.
- B) ACB Gen for the BMP-Transaction in IMS Control Region.
- C) BMP PSB Member (YMI390BM) to be Gen'd.
- D) Assign Certain Class for BMP/IMS Region.
- E) BMP JCL Member (YMI390BP) to be used. (Change Class for BMP Processing.)
- F) BMP Program (YMSMI390) is the Source Member.
  - I) Compile the Revised Source (above) with Required Include Linkage Members. (same as Batch)
  - II) Compile Program to Unique Name "YMI390BM" which is Referenced within BMP JCL
- G) Review JCL Member for Correct Libraries and Parameters.

**Step 13 Backup Data Bases**

Using Your Own Backup JCL, Create a Backup Copy of all the Data Bases.

Backup Source and Load Libraries.

- \* Recommend Image Copy Backup on the Following Data Bases
- \* Which are not Updated: YMI015D, YMI105D, YMI140A, YMI140B, \*
- \* YMI145A, YMI145B, YMI020D, YMI022D, YMI305D, YMI307D, \*
- \* YMI340M, YMI342M, YMI350D, YMI352D, YMI882D, YMI883D.
- \* Recommend DBRC Copy Backup on the Following Data Bases
- \* Which are/may be Updated: YGE010D, YGE010X, YGE010Y, \*
- \* YPA010D, YMI390D, YMI520D, YUM010D, YMI605D, YMI607D, \*
- \* YDUMMY, YGE011D, YGE011X, YGE011Y.

- Step 14**      **Test Procedures (General - Online)**
- Bring Up Test On-line IMS Region.
- User Security Verified (ACF2, IMS Protocol Logon)
- Verify that Test Formats Can be Called Up and Attempt to Update/Inquiry Various Data Bases.
- Logon to System and Verify Password and User IDs.
  - Note - The high level password assigned is "CLIENT".
  - Note - The User ID assigned is USER".

Use the password and user ID above to begin your testing; you may add others per the user manual documentation.

- Use of Master Menu.
- Update/Inquiry Parameter D.B.for Control Records.
- Update/Inquiry Geographic Data.
- Update/Inquiry Secondary Key Point Maintenance.
- Update/Inquiry User Conversion Data.
- Inquiry Mileage processing.
- Inquiry Origin Processing.
- Inquiry ZIP Processing.
- Other (Specify)

- Step 15**      **Test Procedures (General - Batch)**
- Batch IMS Region Available.
- Data Bases Off-line (DBR'd).
- Attempt Inquiry of Mileage (Origin to Destinations).
- Batch JCL Setup (YMI200JC Member - Example).
- Batch Input Origins and Destinations Generated.
- Control Card Used?
- Various Options Used?
- Batch PSB (YMI200) Gen'd.
- JCL Pointing to Correct PSB/DBD Library.
- JCL Pointing to Correct STEPLIB.
- JCL Pointing to Correct Data Base Names.
- Batch programs Compiled to Correct Load Library.

- Other (Specify)
  
- Step 16            Test Procedures (General - BMP)**
- BMP IMS Region Available.
- Data Bases On-line.
- Attempt Inquiry of Mileage (Origin to Destinations).
- BMP JCL Setup (YMI200BP Member - Example).
- BMP Input Origins and Destinations Generated.
- Control Card Used?
- Various Options Used?
- BMP PSB (YMI200BM) Gen'd
- ACB - Transaction Gen'd.
- JCL Pointing to Correct PSB/DBD Library.
- JCL Pointing to Correct STEPLIB.
- BMP program Compiled to Correct Load Library.
- Other (Specify)
- Attempt Processing of YMI390BP (Delete Component Record)
- BMP JCL Setup (YMI390BP Member - Example).
- BMP Input SPLCs (For Deletion).
- BMP PSB (YMI390BM) Gen'd
- ACB - Transaction Gen'd.
- JCL Pointing to Correct PSB/DBD Library.
- JCL Pointing to Correct STEPLIB.
- BMP program Compiled to Correct Load Library.
- Other (Specify)

**Step 17            Set Up Return Screen Mod Name and Length If Needed**

MileMaker will return to a user application screen from the Main Menu when a "6" (return) is entered if a valid MOD name and length is found on a Parameter record. To update the Parameter record, do the following:

- A) Starting at the Main Menu, proceed to Parameter File Inquiry/Update. (Option 4).
- B) Proceed to "MI115 - MileMaker Definitions" (Option E)
- C) Enter an "E".

- D) The screen will display your various definitions. Fill in the MOD name and length.
- E) Return to the Main Menu and enter a "6" to test. If a valid MOD name and length was entered, your screen will be invoked.

Note: This is optional. If not filled in, the Main Menu will simply return to itself. The MOD name field is not edited. The MOD length is edited for numerics. Please use leading zeros in the length field.

**Step 18 Document System Discrepancies**

The user should write up discrepancies as clearly as possible with examples and proper documentation of input, processing, and output-expected results vs actual.

- Unable to Call up Test Format Screens.
- Transactions dont Execute as Documented.
- Programs Abend.
- Documentation Lacking or Incorrect.
- Invalid Responses to Transactions i.e. Wrong Mileages Returned.
- Poor response Time.
- Other (Please Specify)

This concludes the step by step procedure for installation.

## Program Description

	SOURCE NAME	BRIEF DESCRIPTION
1.	YGE010	GEOGRAPHIC UPDATE AND INQUIRY
2.	YMIABBRV	CALLED ROUTINE - CITY/STATE ABBREVIATIONS
3.	YMI001	MASTER MENU PROCESSING
4.	YMI100	MILEAGE INQUIRY EDIT
5.	YMI103	ORIGIN INQUIRY EDIT
6.	YMI105	ZIP INQUIRY EDIT
7.	YMI110	ONLINE INTERFACE MODULE
8.	YMI115	CALLED ROUTINE - MILEAGE PROCESSOR.
9.	YMI120	CALLED ROUTINE - MILEAGE/KEYPT DETERMINATIONS
10.	YMI125	CALLED ROUTINE - MILEAGE CALCULATIONS
11.	YMI140	CALLED ROUTINE - I/O PROCESSOR FOR BOTH GUIDES
12.	YMI605	SECONDARY KEYPOINT MTCE. EDIT
13.	YPA010	PARAMETER UPDATE AND INQUIRY EDIT
14.	YUM010	USER CONVERSION UPDATE AND INQUIRY EDIT
15.	YMI200	BATCH MILEAGE INQUIRY EDIT
16.	YMI201	CALLED ROUTINE - BATCH I/O MODULE
17.	YMI205	CALLED ROUTINE - BATCH ZIP MODULE
18.	YMI210	CALLED ROUTINE - BATCH INTERFACE MODULE
19.	YMI215	CALLED ROUTINE - BATCH MILEAGE PROCESSOR
20.	YMI220	CALLED ROUTINE - BATCH PROCESSING POINTS OTHER THAN KEYPOINTS
21.	YMI225	CALLED ROUTINE - BATCH MILEAGE CALCULATION
22.	YMI230	CALLED ROUTINE - PRINT MILEAGE REPORT (BATCH)
23.	YMI240	CALLED ROUTINE - MILEAGE GUIDE READ
24.	YRANLD1	INITIALIZATION - SECONDARY INDEX FILE(S)
25.	YRANLD2	INITIALIZATION / LOAD - COMMON FILES
26.	YRANLD3	LOAD OF CURRENT GUIDE FILES
27.	YRANLD4	INITIALIZATION / LOAD - PREVIOUS GUIDE FILES
27.	YRANLD6	GEOGRAPHIC MERGE
27.	YRANLD7	CURRENT SECONDARY UPDATE
27.	YRANLD8	PREVIOUS SECONDARY UPDATE
27.	YMERGE7	CURRENT SECONDARY MERGE
27.	YMERGE8	PREVIOUS SECONDARY MERGE
28.	YMSMI20B (BMP)	BMP INQUIRY/EDIT-SAME AS BATCH PGM. (YMI200BM)
29.	YMSMI390 (BMP)	BMP DELETE OF COMPONENT FILE RECORDS (YMI390BM)
27.	YMI116	PREVIOUS GUIDE MILEAGE PROCESSOR (NEW)
27.	YMI121	PREVIOUS GUIDE MILEAGE/KYPT CALCS (NEW)
27.	YMI126	PREVIOUS GUIDE MILEAGE CALCS (NEW)
27.	YMI141	PREVIOUS GUIDE I/O PROCESSOR (NEW)
27.	YMI216	PREVIOUS GUIDE BATCH MILEAGE PROCESSOR (NEW)
27.	YMI221	PREVIOUS GUIDE BATCH MILEAGE/KYPT CALCS (NEW)
27.	YMI226	PREVIOUS GUIDE BATCH MILEAGE CALCS (NEW)
27.	YMI241	PREVIOUS GUIDE BATCH I/O PROCESSOR (NEW)
27.	YMI231	PREVIOUS GUIDE MILEAGE PRINT (NEW)
27.	YRANLDA	INITIALIZES PREVIOUS GUIDE GEO/SEC INDEXES
27.	YRANLDB	LOADS PREVIOUS GUIDE GEO/ZIP FILES
27.	YRANLDC	INITIALIZES YMI605D

## File List (Current Guide)

File Name	File Description	File Type	Record Count	3390 Size(cyls)	Notes
YPA010D	Parameter File	SHISAM	103	1	Control Records (Security/Password)
YMI020D	Locator File	SHISAM	389	2	
YMI305D	Blocker File	SHISAM	51610	290	
YMI605D	Secondary Key Point File	SHISAM	1	1	
YUM010D	User Conversion File	SHISAM	1	1	One Record Initialized
YMI340M	Short Distance Mileage File	SHISAM	4823298	119	
YGE010D	Geographic File	HISAM	154282	75	Secondary Indexes Used
YMI350D	Cross Reference File	SHISAM	6132	1	
YMI390D	Route Component File	SHISAM	1	20	Initialized to one user may create more
YMI520D	Billing File	SHISAM	1	20	Initialized to one user may create more
YMI140B	Mileage File	SHISAM	36792	205	
YMI145B	Traversal File	SHISAM	9868	1	
YMI105D	ZIP Code Master File	SHISAM	984	1	One Record Initialized
YMI015D	ZIP Code Matrix File	SHISAM	907	2	
YDUMMY	Work File	HISAM	1	2	Initialized to one but user will create many by LTERM and TRANS ID
YMI882D	ZIP5 REFERENCE FILE	SHISAM	75166	9	

## File List (Previous Guide)

File Name	File Description	File Type	Record Count	Size(cyls)	Notes
YMI022D	Locator File	SHISAM	389	1	
YMI307D	Blocker File	SHISAM	53576	80	
YMI607D	Secondary Key Point File	SHISAM	200	3	
YMI352D	Cross Reference File	SHISAM	6132	1	
YMI342M	Short Distance Mileage File	SHISAM	3337124	71	
YMI140A	Mileage File	SHISAM	18396	124	
YMI145A	Traversal File	SHISAM	9369	1	
YGE011D	Geographic File	HISAM		75	
YMI883D	ZIP Code File	SHISAM	80747	8	

## File List (Current Guide)

File Name	File Description	Segment Length	Key Length	Notes
YPA010D	Parameter File	80	18	Various IDS "MI115", "FR010", etc. + fields such as Password, Client, "GE010" - State Record
YMI020D	Locator File	2036	6	Longitude - 9(3)V99, Latitude - 9(3)V99 (packed)
YMI305D	Blocker File (variable length)	2207	5	SPLC # (9 pos. packed)
YMI605D	Secondary Key Point File (variable length)	962	5	Internal Key Point (9 pos. packed)
YUM010D	User Conversion File (variable length)	261	25	"Client Code (4 pos.), Type Code ""L"" (1 pos.), User Code (20 pos.)"
YMI340M	Short Distance Mileage File	18	8	Low SPLC (Bin. 4 pos.), High SPLC (Bin. 4 pos.)
YGE010D	Geographic File	124	20	(Primary) State - 2, City/County - 18
	(Alternate Index)	30	5	SPLC # (9 pos. packed) - Secondary Index
	(Alternate Index)	30	5	Internal Key (9 pos. packed)
YMI350D	Cross Reference File	9	3	(Primary) Key Point # (5 pos. packed)
YMI390D	Route Component File	43	12	Internal Keys and Key Point Codes (packed)
YMI520D	Billing File	60	19	Client ID, MO, DY Trans Type (packed low and high SPLCs)
YMI140B	Mileage Guide 18	4090	2	Seq. # (RRDS)
YMI105D	ZIP Code Master File	31	3	Seq. #
YMI015D	ZIP Code Matrix File	2005	5	Seq. # (RRDS)
YDUMMY	Work File	8500	16	LTERM (8 pos.) +
YMI882D	ZIP5 Cross Reference	60	8	ZIP5 or Sequence No.

## File List (Previous Guide)

File Name	File Description	Segment Length	Key Length	Notes
YMI022D	Locator File	2036	6	Longitude - 9(3)\V99! Latitude - 9(3)\V99 (packed)
YMI307D	Blocker File (G18! Variable Length)	907	5	SPLC # (9 pos. packed)
YMI607D	Secondary Key Point File ( Variable Length)	962	5	Internal Key Point (9 pos. packed)
YMI352D	Cross Reference File	9	3	(Primary) Key Point # (5 pos. packed)
YMI342M	Short Distance Mileage File	16	8	Low SPLC (Bin. 4 pos.)! High SPLC (Bin. 4 pos.)
YMI140A	Mileage File	4090	2	Seq. # (RRDS)
YGE011D	Geographic File	124	20	(Primary) State - 2, City/County - 18
YMI883D	ZIP5 Cross Reference	60	8	ZIP5 or Sequence No.

## Parameter File Records

As a new client you will receive a parameter file with all of the essential records; an 'MI115' record, etc. It is suggested that you build your own password and user code records and then delete the ones that were provided by Rand McNally. It is essential that you always have at least one Level 4 Password Record on the Parameter File since access to that file is only permitted to level 4 passwords. You may have as many password and user records as you wish. Further explanation follows:

Parameter File (PA010D) initialized with the records and fields for start of processing of the MILEMAKER SYSTEM.

Member = "YPARMGEO" on Distribution Tape as PANVALET member also.

Five record types:

- (KEY IDS)**
- 1) GE010 (SET UP FOR THE VALID STATE/TERR. CODES)
  - 2) FR001 ----- CONTAINS PASSWORD "CLIENT " -----  
----- CONTAINS REQ. DATA FIELDS -----  
PASSWORD = "CLIENT"  
SECURITY LEV = "4" - ALLOWS MILEMAKER UPDATING  
UPDATE FLAGS/FIELDS
  - 3) FR010 ----- CONTAINS "TDM " USER/CLIENT FIELD -  
----- CONTAINS REQ. DATA FIELDS -----  
  
USER/CLIENT = "TDM"
  - 4) HIVAL (HIGH-VALUES - MUST BE PRESENT TO SYSTEM)

```

5) MI115          ----- CONTAINS REQ. DATA FIELDS -----
   (CONTROL RECORD FIELDS/DATA)
   KEY-ID = "MI115"
   MIN NEAR KYPTS          = "001"
   MAX NEAR KYPTS          = "010"
   SEARCH RADIUS           = "0.10"
   RADIUS INCR             = ".200"
   CURRENT MAX KYPTS SHORTEST          = "06132"
   CURRENT SLTS/BLKS SHORTEST          = "02044"
   CURRENT BLKS/REC SHORTEST          = "00006"
   PREV. MAX KYPTS SHORTEST          = "06132"
   PREV. SLTS/BLKS SHORTEST          = "02044"
   PREV. BLKS/REC SHORTEST          = "00006"
   CURRENT MAX KYPTS PRACTICAL          = "06132"
   CURRENT SLTS/BLKS PRACTICAL          = "01022"
   CURRENT BLKS/REC PRACTICAL          = "00006"
   PREVIOUS GUIDE = "Y" MUST BE Y OR N
                       (Y= BOTH GUIDE 19 AND 18; N=ONLY GUIDE 19)
   EFF DATE = "013108" (GUIDE 18 EFFECTIVE DATE)
   MI390D ROUTE COMP.= "Y" (OPTIONAL - USER TO TURN ON/OFF TO WRITE FILE
   MI520D BILLABLE          = "Y" (OPTIONAL - USER TO TURN ON/OFF TO WRITE FILE
   SHORTEST ROUTE          = "N" (N/A-FOR IMS AT THIS TIME)
   PRACTICAL ROUTE          = "N" (N/A FOR IMS AT THIS TIME)
   TRIP ACCOUNTING          = "N" (N/A FOR IMS AT THIS TIME)

```

For initial installation, all the above fields should be filled in as shown above; user may then revise after checking with Rand McNally systems or customer support areas.

## Modules / Routines General Information ("revisions")

1. **Abbreviation Module (YMIABBRV)** Used by screens which have the city/state as inputted data. It takes the entered data as "Beach" e.g. Palm Beach, and converts it to "BCH" which is the stored field in the geographic data base. This standardization permits ease of key field comparisons between various user-entered data to stored fields.
2. **Access Guide 18 vs Guide 19 (YMI140 On-line) (YMI240 Batch & BMP)**  
 These modules determine which mileage data base (Guide 18 or 19) to read. The guide indicator will show on the display screens. Guide "A" will cause the system to read the data bases for mileage prior to a date of 01/31/08 while Guide "B" indicator will read data bases for its mileages for dates later than 01/31/08. In other words, Guide "A" = prior Guide (Guide 18) while Guide "B" =Current Guide (Guide 19) processing. This indicator is shown on the MI100, MI103 and MI605 screens which allows the user to choose the mileage from that particular Guide (18 or 19).

---

**Note:** Do not confuse this indicator with the following indicator which tells the system whether or not the previous guide is being supported! (Read following section on the proper setting of the parameter indicator for the Previous Guide availability!)

---

3. The point-to-point inquiry screen has been revised to display additional data when the mileage request involves a calculated point. In that case the surrounding points which block the requested point(s) will also be shown. These additional points will better highlight the mileages and distances which have been determined/calculated for the particular request.

## **Files / Data Bases General Information**

1. **Route Component File (YMI390D)** This is a file which is optional as far as whether or not the user will write records to it. This data base, when written to, will contain the mileage on point to point inquiries that require processing node/surrounding blockers. Once this file has a record written to it for a request, then that record will be read for any future mileages to the same pt. to pt. request.
2. An additional alternate index has been added to the Geographic File (Internal Key) Which will be used throughout the programs instead of the 'SPLC' code. This alternate index field will be the key field used in accessing all the other files. YGE010DY is the identifier for this data. This Internal Key field may not be changed at any time. The 'city/state' key field may be changed in the GE010D FILE. The 'SPLC' key field may also be changed in the GE010D file.
3. **Billable File (MI520D)** This is an optional file which can be turned on for tracking of user activity. The parameter screen contains the indicator to turn on/off depending upon the user's needs. Mileages are not affected by this file.

This file is used for the tracking and audit of client activity. For each mileage inquiry a new record is written or an existing record is updated for data as date, time, user ID., and data related to what mileage was requested by the user. These records can be used for billing and statistical tracking of customer pt-pt, origin, ZIP and Geographic Browse activity. This data base is initialized to one record and thereafter the user will monitor and control this file (volume of data base dependent upon use of system).



---

## Chapter Contents

- BATCH MILEAGE/ZIP INQUIRY ..... 52**
  - Batch System Description ..... 52**
  - Record Description: Input Control Record ..... 53**
  - Record Description: Format One Input Record ..... 54**
  - Record Description: Format Two Input Record ..... 55**
  - Record Description: Tape Output Control Record ..... 56**
  - Record Description: Tape Output Copyright Records ..... 57**
  - Record Description: Tape Output Record (Format One) ..... 58**
  - Record Description: Tape Output Record (Format Two) ..... 59**
  - Report Output Description ..... 60**
  - Error Messages ..... 61**

---

## Batch Mileage/ZIP Inquiry

The mileage inquiry program reads input containing points for mileage, ZIP3, or ZIP5 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.

## Batch System Description

The Batch System processes mileage inquiries using the same data bases as on-line. The programs YMSMI200 through YMSMI240, PSB YMI200, along with sample JCL members YMI200JC, YMI200J1, YMI200J2 are supplied by the user(s) building this separate, stand-alone IMS system.

An interface program, YMSMI210, is part of the batch system which will allow the user to input batch mileage requests from their system into MileMaker and have them returned to the user's system via the appropriate linkage areas. (Refer to the interface document for the system specifications - same as online.)

The BMP (Batch Message Processing) program, YMSMI20B, and PSB YMI200BM along with sample JCL (YMI200BP) are also furnished. This processing operates in the same manner as the batch-type requests.

## 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	1	SPLC indicator	Valid parameter values for SPLC are "Y" =Yes, and "N" = No. If left blank, "N" will be assumed. "Y" causes the SPLC to appear on output tape records.
9	50	Client name	Mandatory
59	*	Filler	Spaces.

\*This field is 22 bytes long for format one and 59 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 indicator	Must be 'O' for origin or 'D' for destination
2	3	ZIP3 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. Military SPLCs for Current Guide may be six digits; the 2XX sub code SPLC series would be used for positions 7 through 9.
2	2	State	Two character state/province code.
2	5	ZIP5 Code	5-digit ZIP Code must be five digits.
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...5...10...15...20...25	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. Military SPLCs for Current Guide may be six digits; the 2XX sub code SPLC series would be used for positions 7 through 9.
1	2	Origin state	Two character state or province code.
1	5	Origin ZIP5 Code	5-digit ZIP Code must be five digits.
3	40	Origin city	City name or city with a county qualifier preceded by a comma. See note on page 54 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	2	Destination state	Two character state/province code.
61	40	Destination city	City name or city with county qualifier preceded by a comma. See note on page 54 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	1	SPLC indicator	Valid parameter values for SPLC are "Y" =Yes, and "N" = No. If left blank, "N" will be assumed. "Y" causes the SPLC to appear on output tape records.
9	50	Client name	As it appears on the input control record.
59	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 input.
69	4	Mileage	Four digit display numeric mileage appears on destination records. On origin records this field contains zeros.
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 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.
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:

<b>S</b>	The mileage is a stored mileage
<b>C</b>	Some portion of the mileage has been calculated because either the origin or the destination, or both, is not on the highway network.
<b>E</b>	There was an error encountered in obtaining the mileage. Consult your Customer Support Representative.
<b>N</b>	Unable to determine mileage (Data base(s) missing, invalid processing, etc.)
*	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.

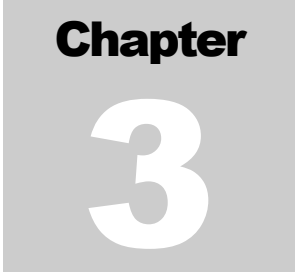
**No Errors Detected**

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.

# INTERFACE DOCUMENTATION



---

## Chapter Contents

**MILEAGE/ZIP INTERFACE ..... 64**  
    **Mileage/ZIP Interface Requirements..... 64**  
    **Field Definitions ..... 64**  
**LAYOUT OF STORAGE AREA (ONLINE)..... 68**  
**LAYOUT OF STORAGE AREA (BATCH)..... 69**  
**SAMPLE IMS BMP INTERFACE CODE SEGMENTS ..... 70**

---

# Mileage/ZIP Interface

## Mileage/ZIP Interface Requirements

User applications may obtain mileages by using the provided batch or online interface modules. State/City, SPLC, 3-digit ZIP Code, or 5-digit ZIP Code may be used. Standard linkage conventions are used.

The user, through his application calling program, can receive mileages from MileMaker modules. Two types of requests are available; origin and mileage. An origin request consists of a single origin designated by City/State or SPLC and a maximum of 39 destination points. The values returned are the mileages from the origin point to the various points. A mileage request consists of a maximum of 40 points designated by City/State, SPLC, or ZIP Code. (only mileage requests are supported when using ZIP Codes.) The mileages returned are from each point to the next point.

The user application program must define certain fields in program storage and establish linkage to these fields when MileMaker modules are called. Examples of this are found in source members YMI110LN (online) and YMI210LN (batch). The user program PSB must also be updated to include MileMaker PCBs. The additional PCBs are found in source members YMI110LN and YMI210LN.

### **ONLINE**

The YMI110 program is linked to and called by the user's application program.

### **BATCH**

The YMI210 program is linked to and called by the user's application program.

## 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 ZIP3 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 Code 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 occurred 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:

<b>NF</b>	At least one point was not found on the Geographic or User Conversion files.
<b>NP</b>	Previous Guide was requested but is not supported.
<b>NE</b>	The number of entries field was invalid -- either it was not numeric, less than 2 or greater than 40.
<b>ST</b>	At least one entry has an 'E' status value in the MI-STATUS field. (see MI-STATUS values on page 66.)
<b>CE</b>	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.
<b>Spaces</b>	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 Code 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 Code 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 16, followed by county code in positions 17 and 18. 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 Code 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 of the entry point. This field must be a five-digit number. Canadian ZIPs are not supported. If city/state or SPLC is used, MI-ZIP5 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:

<b>01</b>	City name needs county qualifier
<b>02</b>	City/state not found (online)
<b>03</b>	File error has occurred (See MI-ERROR-FILE to determine which file is involved)
<b>05</b>	City/state not found (batch)

**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 occurred 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:

<b>S</b>	Stored mileage
<b>C</b>	Calculated mileage
<b>E</b>	Mileage error encountered
<b>N</b>	Unable to determine mileage (data base(s) missing, invalid processing etc.)

**MI-MILEAGE.** This packed numeric field should be initialized to zeros. If an error is encountered, zeros will be returned. The mileage (whether point to oint or origin to destination) will be returned to this field.

## MI-MILEAGE FIELD-VALUES RETURNED

**Point to Point Request** as point #1 to point #2, then point #2 to point #3 and so on :

OCCURS ENTRY	#1 = POINT#1	= MI-STATE-CITY (1)
VALID MILEAGE BETWEEN	POINT #1 AND POINT #2	= MI-MILEAGE (1)
OCCURS ENTRY	#2 = POINT#2	= MI-STATE-CITY (2)
VALID MILEAGE BETWEEN	POINT #2 AND POINT #3	= MI-MILEAGE (2)
OCCURS ENTRY	#3 = POINT#3	= MI-STATE-CITY (3)
TOTAL TRIP MILEAGE		= MI-MILEAGE (3)

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) mileage occurs field.

### Origin To Destinations Request

OCCURS ENTRY	#1 = POINT#1	= MI-STATE-CITY (1) Represents origin point
	ZEROS	= MI-MILEAGE (1)
OCCURS ENTRY	#2 = POINT#2	= MI-STATE-CITY (2) = destination
MILEAGE BETWEEN POINTS	#1 AND #2	= MI-MILEAGE (2)
OCCURS ENTRY	#3 = POINT#3	= MI-STATE-CITY (3) = another destination
MILEAGE BETWEEN POINTS	#1 AND #3	= MI-MILEAGE (3)

AND SO ON FOR A MAXIMUM OF 40 OCCUR ENTRIES

The valid mileage will be returned in each **DESTINATION** point entry. In the case of an origin request, there is **NO** total mileage returned for the group of points requested.

**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.

---

## Layout of Storage Area (Online)

```
05 MI-POINT-DATA
  10 MI-REQUEST-TYPE          PIC X(1) .
  10 MI-ERROR-TYPE           PIC X(2) .
  10 MI-MILEAGE-IND           PIC X(1) .
  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 .
```

---

**Note:** Refer to source member YMI110LN for examples of linkage fields, call statements, and PCB entries.

---

---

## Layout of Storage Area (Batch)

```
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) .
  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 .
```

---

**Note:** Refer to source member YMI210LN for examples of linkage fields, call statements, and PCB entries.

---

---

## Sample IMS BMP Interface Code Segments

```
01 LTERM-PCB.
   05 FILLER PIC X(10).
   05 TPSTATUS PIC X(02).
   05 FILLER PIC X(20).
01 PA010D-PCB.
   05 PA010D-DBDNAME PIC X(08).
   05 PA010D-LEV-NUM PIC 9(02).
   05 PA010D-PCB-STAT PIC X(02).
   05 PA010D-POC-OPT PIC X(04).
   05 PA010D-JCB-ADDR PIC X(04).
   05 PA010D-SEG-NAME PIC X(08).
   05 PA010D-KEY-LEN PIC S9(05) COMP.
   05 PA010D-NUM-SEGS PIC S9(05) COMP.
   05 PA010D-FEEDBACK PIC X(18).
   SKIP1
01 MI305D-PCB.
   05 MI305D-DBDNAME PIC X(08).
   05 MI305D-LEV-NUM PIC 9(02).
   05 MI305D-PCB-STAT PIC X(02).
   05 MI305D-POC-OPT PIC X(04).
   05 MI305D-JCB-ADDR PIC X(04).
   05 MI305D-SEG-NAME PIC X(08).
   05 MI305D-KEY-LEN PIC S9(05) COMP.
   05 MI305D-NUM-SEGS PIC S9(05) COMP.
   05 MI305D-FEEDBACK PIC X(5).
   SKIP1
01 MI350D-PCB.
   05 MI350D-DBDNAME PIC X(08).
   05 MI350D-LEV-NUM PIC 9(02).
   05 MI350D-PCB-STAT PIC X(02).
   05 MI350D-POC-OPT PIC X(04).
   05 MI350D-JCB-ADDR PIC X(04).
   05 MI350D-SEG-NAME PIC X(08).
   05 MI350D-KEY-LEN PIC S9(05) COMP.
   05 MI350D-NUM-SEGS PIC S9(05) COMP.
   05 MI350D-FEEDBACK PIC X(3).
   SKIP1
01 MI605D-PCB.
   05 MI605D-DBDNAME PIC X(08).
   05 MI605D-LEV-NUM PIC 9(02).
   05 MI605D-PCB-STAT PIC X(02).
   05 MI605D-POC-OPT PIC X(04).
   05 MI605D-JCB-ADDR PIC X(04).
   05 MI605D-SEG-NAME PIC X(08).
   05 MI605D-KEY-LEN PIC S9(05) COMP.
   05 MI605D-NUM-SEGS PIC S9(05) COMP.
   05 MI605D-FEEDBACK PIC X(5).
   SKIP1
01 MI340M-PCB.
   05 MI340M-DBDNAME PIC X(08).
   05 MI340M-LEV-NUM PIC 9(02).
   05 MI340M-PCB-STAT PIC X(02).
   05 MI340M-POC-OPT PIC X(04).
   05 MI340M-JCB-ADDR PIC X(04).
   05 MI340M-SEG-NAME PIC X(08).
   05 MI340M-KEY-LEN PIC S9(05) COMP.
   05 MI340M-NUM-SEGS PIC S9(05) COMP.
   05 MI340M-FEEDBACK PIC X(8).
   SKIP1
01 MI140A-PCB.
   05 MI140A-DBDNAME PIC X(08).
   05 MI140A-LEV-NUM PIC 9(02).
```

05 MI140A-PCB-STAT	PIC X(02).	
05 MI140A-POC-OPT	PIC X(04).	
05 MI140A-JCB-ADDR	PIC X(04).	
05 MI140A-SEG-NAME	PIC X(08).	
05 MI140A-KEY-LEN	PIC S9(05)	COMP.
05 MI140A-NUM-SEGS	PIC S9(05)	COMP.
05 MI140A-FEEDBACK	PIC X(5).	
SKIPI		
01 MI140B-PCB.		
05 MI140B-DBDNAME	PIC X(08).	
05 MI140B-LEV-NUM	PIC 9(02).	
05 MI140B-PCB-STAT	PIC X(02).	
05 MI140B-POC-OPT	PIC X(04).	
05 MI140B-JCB-ADDR	PIC X(04).	
05 MI140B-SEG-NAME	PIC X(08).	
05 MI140B-KEY-LEN	PIC S9(05)	COMP.
05 MI140B-NUM-SEGS	PIC S9(05)	COMP.
05 MI140B-FEEDBACK	PIC X(5).	
SKIPI		
01 MI145A-PCB.		
05 MI145A-DBDNAME	PIC X(08).	
05 MI145A-LEV-NUM	PIC 9(02).	
05 MI145A-PCB-STAT	PIC X(02).	
05 MI145A-POC-OPT	PIC X(04).	
05 MI145A-JCB-ADDR	PIC X(04).	
05 MI145A-SEG-NAME	PIC X(08).	
05 MI145A-KEY-LEN	PIC S9(05)	COMP.
05 MI145A-NUM-SEGS	PIC S9(05)	COMP.
05 MI145A-FEEDBACK	PIC X(4).	
SKIPI		
01 MI145B-PCB.		
05 MI145B-DBDNAME	PIC X(08).	
05 MI145B-LEV-NUM	PIC 9(02).	
05 MI145B-PCB-STAT	PIC X(02).	
05 MI145B-POC-OPT	PIC X(04).	
05 MI145B-JCB-ADDR	PIC X(04).	
05 MI145B-SEG-NAME	PIC X(08).	
05 MI145B-KEY-LEN	PIC S9(05)	COMP.
05 MI145B-NUM-SEGS	PIC S9(05)	COMP.
05 MI145B-FEEDBACK	PIC X(4).	
SKIPI		
01 MI020D-PCB.		
05 MI020D-DBDNAME	PIC X(08).	
05 MI020D-LEV-NUM	PIC 9(02).	
05 MI020D-PCB-STAT	PIC X(02).	
05 MI020D-POC-OPT	PIC X(04).	
05 MI020D-JCB-ADDR	PIC X(04).	
05 MI020D-SEG-NAME	PIC X(08).	
05 MI020D-KEY-LEN	PIC S9(05)	COMP.
05 MI020D-NUM-SEGS	PIC S9(05)	COMP.
05 MI020D-FEEDBACK	PIC X(6).	
SKIPI		
01 GE010D-PCB.		
05 GE010D-DBDNAME	PIC X(08).	
05 GE010D-LEV-NUM	PIC 9(02).	
05 GE010D-PCB-STAT	PIC X(02).	
05 GE010D-POC-OPT	PIC X(04).	
05 GE010D-JCB-ADDR	PIC X(04).	
05 GE010D-SEG-NAME	PIC X(08).	
05 GE010D-KEY-LEN	PIC S9(05)	COMP.
05 GE010D-NUM-SEGS	PIC S9(05)	COMP.
05 GE010D-FEEDBACK	PIC X(20).	
SKIPI		
01 GE010X-PCB.		
05 GE010X-DBDNAME	PIC X(08).	
05 GE010X-LEV-NUM	PIC 9(02).	
05 GE010X-PCB-STAT	PIC X(02).	
05 GE010X-POC-OPT	PIC X(04).	
05 GE010X-JCB-ADDR	PIC X(04).	

05	GE010X-SEG-NAME	PIC X(08).	
05	GE010X-KEY-LEN	PIC S9(05)	COMP.
05	GE010X-NUM-SEGS	PIC S9(05)	COMP.
05	GE010X-FEEDBACK	PIC X(5).	
	SKIPI		
01	GE010Y-PCB.		
05	GE010Y-DBDNAME	PIC X(08).	
05	GE010Y-LEV-NUM	PIC 9(02).	
05	GE010Y-PCB-STAT	PIC X(02).	
05	GE010Y-POC-OPT	PIC X(04).	
05	GE010Y-JCB-ADDR	PIC X(04).	
05	GE010Y-SEG-NAME	PIC X(08).	
05	GE010Y-KEY-LEN	PIC S9(05)	COMP.
05	GE010Y-NUM-SEGS	PIC S9(05)	COMP.
05	GE010Y-FEEDBACK	PIC X(5).	
	SKIPI		
01	UM010D-PCB.		
05	UM010D-DBDNAME	PIC X(08).	
05	UM010D-LEV-NUM	PIC 9(02).	
05	UM010D-PCB-STAT	PIC X(02).	
05	UM010D-POC-OPT	PIC X(04).	
05	UM010D-JCB-ADDR	PIC X(04).	
05	UM010D-SEG-NAME	PIC X(08).	
05	UM010D-KEY-LEN	PIC S9(05)	COMP.
05	UM010D-NUM-SEGS	PIC S9(05)	COMP.
05	UM010D-FEEDBACK	PIC X(25).	
	SKIPI		
01	MI390D-PCB.		
05	MI390D-DBDNAME	PIC X(08).	
05	MI390D-LEV-NUM	PIC 9(02).	
05	MI390D-PCB-STAT	PIC X(02).	
05	MI390D-POC-OPT	PIC X(04).	
05	MI390D-JCB-ADDR	PIC X(04).	
05	MI390D-SEG-NAME	PIC X(08).	
05	MI390D-KEY-LEN	PIC S9(05)	COMP.
05	MI390D-NUM-SEGS	PIC S9(05)	COMP.
05	MI390D-FEEDBACK	PIC X(12).	
	SKIPI		
01	MI520D-PCB.		
05	MI520D-DBDNAME	PIC X(08).	
05	MI520D-LEV-NUM	PIC 9(02).	
05	MI520D-PCB-STAT	PIC X(02).	
05	MI520D-POC-OPT	PIC X(04).	
05	MI520D-JCB-ADDR	PIC X(04).	
05	MI520D-SEG-NAME	PIC X(08).	
05	MI520D-KEY-LEN	PIC S9(05)	COMP.
05	MI520D-NUM-SEGS	PIC S9(05)	COMP.
05	MI520D-FEEDBACK	PIC X(19).	
	SKIPI		
01	MI307D-PCB.		
05	MI307D-DBDNAME	PIC X(08).	
05	MI307D-LEV-NUM	PIC 9(02).	
05	MI307D-PCB-STAT	PIC X(02).	
05	MI307D-POC-OPT	PIC X(04).	
05	MI307D-JCB-ADDR	PIC X(04).	
05	MI307D-SEG-NAME	PIC X(08).	
05	MI307D-KEY-LEN	PIC S9(05)	COMP.
05	MI307D-NUM-SEGS	PIC S9(05)	COMP.
05	MI307D-FEEDBACK	PIC X(05).	
	SKIPI		
01	MI352D-PCB.		
05	MI352D-DBDNAME	PIC X(08).	
05	MI352D-LEV-NUM	PIC 9(02).	
05	MI352D-PCB-STAT	PIC X(02).	
05	MI352D-POC-OPT	PIC X(04).	
05	MI352D-JCB-ADDR	PIC X(04).	
05	MI352D-SEG-NAME	PIC X(08).	
05	MI352D-KEY-LEN	PIC S9(05)	COMP.
05	MI352D-NUM-SEGS	PIC S9(05)	COMP.

05 MI352D-FEEDBACK	PIC X(3).	
SKIPI		
01 MI607D-PCB.		
05 MI607D-DBDNAME	PIC X(08).	
05 MI607D-LEV-NUM	PIC 9(02).	
05 MI607D-PCB-STAT	PIC X(02).	
05 MI607D-POC-OPT	PIC X(04).	
05 MI607D-JCB-ADDR	PIC X(04).	
05 MI607D-SEG-NAME	PIC X(08).	
05 MI607D-KEY-LEN	PIC S9(05)	COMP.
05 MI607D-NUM-SEGS	PIC S9(05)	COMP.
05 MI607D-FEEDBACK	PIC X(5).	
SKIPI		
01 MI342M-PCB.		
05 MI342M-DBDNAME	PIC X(08).	
05 MI342M-LEV-NUM	PIC 9(02).	
05 MI342M-PCB-STAT	PIC X(02).	
05 MI342M-POC-OPT	PIC X(04).	
05 MI342M-JCB-ADDR	PIC X(04).	
05 MI342M-SEG-NAME	PIC X(08).	
05 MI342M-KEY-LEN	PIC S9(05)	COMP.
05 MI342M-NUM-SEGS	PIC S9(05)	COMP.
05 MI342M-FEEDBACK	PIC X(08).	
SKIPI		
01 MI022D-PCB.		
05 MI022D-DBDNAME	PIC X(08).	
05 MI022D-LEV-NUM	PIC 9(02).	
05 MI022D-PCB-STAT	PIC X(02).	
05 MI022D-POC-OPT	PIC X(04).	
05 MI022D-JCB-ADDR	PIC X(04).	
05 MI022D-SEG-NAME	PIC X(08).	
05 MI022D-KEY-LEN	PIC S9(05)	COMP.
05 MI022D-NUM-SEGS	PIC S9(05)	COMP.
05 MI022D-FEEDBACK	PIC X(06).	
SKIPI		
01 MI015D-PCB.		
05 MI015D-DBDNAME	PIC X(08).	
05 MI015D-LEV-NUM	PIC 9(02).	
05 MI015D-PCB-STAT	PIC X(02).	
05 MI015D-POC-OPT	PIC X(04).	
05 MI015D-JCB-ADDR	PIC X(04).	
05 MI015D-SEG-NAME	PIC X(08).	
05 MI015D-KEY-LEN	PIC S9(05)	COMP.
05 MI015D-NUM-SEGS	PIC S9(05)	COMP.
05 MI015D-FEEDBACK	PIC X(05).	
SKIPI		
01 MI105D-PCB.		
05 MI105D-DBDNAME	PIC X(08).	
05 MI105D-LEV-NUM	PIC 9(02).	
05 MI105D-PCB-STAT	PIC X(02).	
05 MI105D-POC-OPT	PIC X(04).	
05 MI105D-JCB-ADDR	PIC X(04).	
05 MI105D-SEG-NAME	PIC X(08).	
05 MI105D-KEY-LEN	PIC S9(05)	COMP.
05 MI105D-NUM-SEGS	PIC S9(05)	COMP.
05 MI105D-FEEDBACK	PIC X(03).	
SKIPI		
01 MI882D-PCB.		
05 MI882D-DBDNAME	PIC X(08).	
05 MI882D-LEV-NUM	PIC 9(02).	
05 MI882D-PCB-STAT	PIC X(02).	
05 MI882D-POC-OPT	PIC X(04).	
05 MI882D-JCB-ADDR	PIC X(04).	
05 MI882D-SEG-NAME	PIC X(08).	
05 MI882D-KEY-LEN	PIC S9(05)	COMP.
05 MI882D-NUM-SEGS	PIC S9(05)	COMP.
05 MI882D-FEEDBACK	PIC X(03).	
SKIPI		
01 GE011D-PCB.		

05	GE011D-DBDNAME	PIC X(08).	
05	GE011D-LEV-NUM	PIC 9(02).	
05	GE011D-PCB-STAT	PIC X(02).	
05	GE011D-POC-OPT	PIC X(04).	
05	GE011D-JCB-ADDR	PIC X(04).	
05	GE011D-SEG-NAME	PIC X(08).	
05	GE011D-KEY-LEN	PIC S9(05)	COMP.
05	GE011D-NUM-SEGS	PIC S9(05)	COMP.
05	GE011D-FEEDBACK	PIC X(20).	
	SKIPI		
01	GE011X-PCB.		
05	GE011X-DBDNAME	PIC X(08).	
05	GE011X-LEV-NUM	PIC 9(02).	
05	GE011X-PCB-STAT	PIC X(02).	
05	GE011X-POC-OPT	PIC X(04).	
05	GE011X-JCB-ADDR	PIC X(04).	
05	GE011X-SEG-NAME	PIC X(08).	
05	GE011X-KEY-LEN	PIC S9(05)	COMP.
05	GE011X-NUM-SEGS	PIC S9(05)	COMP.
05	GE011X-FEEDBACK	PIC X(5).	
	SKIPI		
01	GE011Y-PCB.		
05	GE011Y-DBDNAME	PIC X(08).	
05	GE011Y-LEV-NUM	PIC 9(02).	
05	GE011Y-PCB-STAT	PIC X(02).	
05	GE011Y-POC-OPT	PIC X(04).	
05	GE011Y-JCB-ADDR	PIC X(04).	
05	GE011Y-SEG-NAME	PIC X(08).	
05	GE011Y-KEY-LEN	PIC S9(05)	COMP.
05	GE011Y-NUM-SEGS	PIC S9(05)	COMP.
05	GE011Y-FEEDBACK	PIC X(5).	
	SKIPI		
01	MI883D-PCB.		
05	MI883D-DBDNAME	PIC X(08).	
05	MI883D-LEV-NUM	PIC 9(02).	
05	MI883D-PCB-STAT	PIC X(02).	
05	MI883D-POC-OPT	PIC X(04).	
05	MI883D-JCB-ADDR	PIC X(04).	
05	MI883D-SEG-NAME	PIC X(08).	
05	MI883D-KEY-LEN	PIC S9(05)	COMP.
05	MI883D-NUM-SEGS	PIC S9(05)	COMP.
05	MI883D-FEEDBACK	PIC X(03).	
	SKIPI		

ONLINE LAYOUT OF STORAGE AREA IN MILEMAKER PROGRAMS  
TO STORE ORIGIN AND DESTINATION POINTS DATA.

```

05 MI-POINT-DATA
  10 MI-REQUEST-TYPE PIC X(1).
  10 MI-ERROR-TYPE PIC X(2).
  10 MI-MILEAGE-IND PIC X(1).
  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.

SAMPLE COBOL ENTRY STATEMENT FOR DLITCBL  
ENTRY 'DLITCBL' USING LTERM-PCB

PA010D-PCB  
MI305D-PCB  
MI350D-PCB  
MI605D-PCB  
MI340M-PCB  
MI140A-PCB  
MI140B-PCB  
MI145A-PCB  
MI145B-PCB  
MI020D-PCB  
GE010D-PCB  
GE010X-PCB  
GE010Y-PCB  
UM010D-PCB  
MI390D-PCB  
MI520D-PCB  
MI307D-PCB  
MI352D-PCB  
MI607D-PCB  
MI342M-PCB  
MI022D-PCB  
MI015D-PCB  
MI105D-PCB  
MI882D-PCB  
GE011D-PCB  
GE011X-PCB  
GE011Y-PCB  
MI883D-PCB

SAMPLE CALLING STATEMENT WITH PARAMETER LIST

```
CALL 'YMI110' USING PA010D-PCB
MI305D-PCB
MI350D-PCB
MI605D-PCB
MI340M-PCB
MI140A-PCB
MI140B-PCB
MI145A-PCB
MI145B-PCB
MI020D-PCB
GE010D-PCB
GE010X-PCB
GE010Y-PCB
UM010D-PCB
MI390D-PCB
MI520D-PCB
MI307D-PCB
MI352D-PCB
MI607D-PCB
MI342M-PCB
MI022D-PCB
MI015D-PCB
MI105D-PCB
MI-AREA
MI882D-PCB
GE011D-PCB
GE011X-PCB
GE011Y-PCB
MI883D-PCB
```

EXAMPLE OF PCB S TO BE ADDED TO USERS PSB.  
GUIDE 19.

```
PCB TYPE=DB, DBDNAME=YPA010D, PROCOPT=A, KEYLEN=18
SENSESEGNAME=YPA010DS, PARENT=0
PCB TYPE=DB, DBDNAME=YMI305D, PROCOPT=A, KEYLEN=5
SENSESEGNAME=YMI305DS, PARENT=0
PCB TYPE=DB, DBDNAME=YMI350D, PROCOPT=A, KEYLEN=3
SENSESEGNAME=YMI350DS, PARENT=0
PCB TYPE=DB, DBDNAME=YMI605D, PROCOPT=A, KEYLEN=5
SENSESEGNAME=YMI605DS, PARENT=0
PCB TYPE=DB, DBDNAME=YMI340M, PROCOPT=A, KEYLEN=8
SENSESEGNAME=YMI340MS, PARENT=0
PCB TYPE=DB, DBDNAME=YMI140A, PROCOPT=A, KEYLEN=2
SENSESEGNAME=YMI140AS, PARENT=0
PCB TYPE=DB, DBDNAME=YMI140B, PROCOPT=A, KEYLEN=2
SENSESEGNAME=YMI140BS, PARENT=0
PCB TYPE=DB, DBDNAME=YMI145A, PROCOPT=A, KEYLEN=4
SENSESEGNAME=YMI145AS, PARENT=0
PCB TYPE=DB, DBDNAME=YMI145B, PROCOPT=A, KEYLEN=4
SENSESEGNAME=YMI145BS, PARENT=0
PCB TYPE=DB, DBDNAME=YMI020D, PROCOPT=A, KEYLEN=6
SENSESEGNAME=YMI020DS, PARENT=0
PCB TYPE=DB, DBDNAME=YGE010D, PROCOPT=A, KEYLEN=20
SENSESEGNAME=YGE010DS, PARENT=0
PCB TYPE=DB, DBDNAME=YGE010D, PROCOPT=G, KEYLEN=5, PROCSEQ=YGE010X
SENSESEGNAME=YGE010DS, PARENT=0
PCB TYPE=DB, DBDNAME=YGE010D, PROCOPT=A, KEYLEN=5, PROCSEQ=YGE010Y
SENSESEGNAME=YGE010DS, PARENT=0
PCB TYPE=DB, DBDNAME=YUM010D, PROCOPT=A, KEYLEN=25
SENSESEGNAME=YUM010DS, PARENT=0
PCB TYPE=DB, DBDNAME=YMI390D, PROCOPT=A, KEYLEN=12
SENSESEGNAME=YMI390DS, PARENT=0
PCB TYPE=DB, DBDNAME=YMI520D, PROCOPT=A, KEYLEN=19
SENSESEGNAME=YMI520DS, PARENT=0
PCB TYPE=DB, DBDNAME=YMI307D, PROCOPT=A, KEYLEN=5
SENSESEGNAME=YMI307DS, PARENT=0
PCB TYPE=DB, DBDNAME=YMI352D, PROCOPT=A, KEYLEN=3
```

SENSENAME=YMI352DS, PARENT=0  
PCB TYPE=DB, DBDNAME=YMI607D, PROCOPT=A, KEYLEN=5  
SENSENAME=YMI607DS, PARENT=0  
PCB TYPE=DB, DBDNAME=YMI342M, PROCOPT=A, KEYLEN=8  
SENSENAME=YMI342MS, PARENT=0  
PCB TYPE=DB, DBDNAME=YMI022D, PROCOPT=A, KEYLEN=6  
SENSENAME=YMI022DS, PARENT=0  
PCB TYPE=DB, DBDNAME=YMI015D, PROCOPT=A, KEYLEN=5  
SENSENAME=YMI015DS, PARENT=0  
PCB TYPE=DB, DBDNAME=YMI105D, PROCOPT=A, KEYLEN=3  
SENSENAME=YMI105DS, PARENT=0  
PCB TYPE=DB, DBDNAME=YMI882D, PROCOPT=A, KEYLEN=8  
SENSENAME=YMI882DS, PARENT=0  
PCB TYPE=DB, DBDNAME=YGE011D, PROCOPT=A, KEYLEN=20  
SENSENAME=YGE011DS, PARENT=0  
PCB TYPE=DB, DBDNAME=YGE011D, PROCOPT=G, KEYLEN=5, PROCSEQ=YGE011X  
SENSENAME=YGE011DS, PARENT=0  
PCB TYPE=DB, DBDNAME=YGE011D, PROCOPT=A, KEYLEN=5, PROCSEQ=YGE011Y  
SENSENAME=YGE011DS, PARENT=0  
PCB TYPE=DB, DBDNAME=YMI883D, PROCOPT=A, KEYLEN=8  
SENSENAME=YMI883DS, PARENT=0



# COMPONENT FILE DELETION



---

## Chapter Contents

**COMPONENT FILE DELETION PROGRAM..... 80**  
    **Description of YMSMI390 Program ..... 80**  
    **Sample JCL for YMI390BP ..... 81**  
    **Input SPLC File Record Format ..... 81**  
    **Display Examples Produced by YMSMI390 Program..... 82**

---

# Component File Deletion Program

## Description of YMSMI390 Program

The Component File Delete Program offers a method of performing mass deletions from the Component File (YMI390D). This program will normally be run only at the direction of Rand McNally. It is used to delete all YMI390D 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 YMI390D file and start with an initialized 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 MI605D record for a Secondary Keypoint.

This is a BMP which must be processed while the data bases are online and, preferably, while there is minimal user activity on the system. Accordingly, schedule the job to run over night or at off times so as not to impact the users on the system. The processing of the job takes extended check points, and various display messages are printed for information as well as if any problems occur with the YMI390D data base calls.



## Display Examples Produced by YMSMI390 Program

```
**YMI390BM PGM DISPLAY**
VALID SPLC SUBMITTED=999999999
**YMI390BM PGM DISPLAY**
INVALID SPLC SUBMITTED=999999999
RECORD BYPASSED - PROCESSING CONTINUES
*****
YMI390BM PROGRAM DISPLAY COUNTS
NUMBER MI390D RECORDS READ = 999999
NUMBER MI390D RECORDS DELETED = 999999
*****
OTHER DISPLAYS THAT WILL APPEAR IF APPLICABLE
*****
YMI390BM PGM DISPLAY
MORE THAN FIVE SPLCS SUBMITTED
SPLC BYPASSED = 999999999
PROCESSING CONTINUES
*****
YMI390BM PGM DISPLAY
NO INPUT RECORDS READ
JOB ENDED - NO PROCESSING DONE
*****
```