



RESOURCE AND PATIENT MANAGEMENT SYSTEM

BMXNet40

ADO.NET Data Access and Connectivity Utilities for RPMS
Including WinForm and EHR Integration Frameworks

(BMX)

Technical Manual

Version 4.0
November 2019

Office of Information Technology
Division of Information Technology

Table of Contents

1.0	Introduction.....	1
2.0	Orientation	2
3.0	Implementation and Maintenance	3
3.1	System Requirements	3
3.2	Package-Wide Variables	3
3.3	Security Keys.....	3
4.0	Routine Descriptions	4
4.1	Routines with Description	4
5.0	Files and Tables.....	7
5.1	File List	7
5.2	File Access	7
5.3	Cross References	7
5.4	Table File.....	10
5.5	Client Assembly Information	12
6.0	External Relations	13
6.1	Callable Routines.....	13
6.2	Published Entry Points.....	15
6.3	Exported Options	15
6.4	Client Components	16
7.0	Internal Relations	17
8.0	Archiving and Purging	18
9.0	Generating Online Documentation	19
9.1	System Documentation.....	19
9.1.1	%INDEX	19
9.1.2	Inquire Option.....	20
9.1.3	Print Option File.....	20
9.1.4	List File Attributes	20
9.1.5	Online Help.....	21
10.0	SAC Requirements and Exemptions	22
	Glossary.....	23
	Acronym List	27
	Contact Information	28

Revision History

Version	Date	Author	Section	Page Number	Summary of Change
1.0	06/16/2005				Initial 1.0 Version
2.0	11/06/2006		3.1		Added routine BMXPO; associated updates
2.0	11/06/2006		4.1, 4.2		Added file BMX GUI REPORTS; associated updates
2.0	03/07/2007 03/14/2007		2.1		Changed system requirements from "Cache 5.1" to 5.0.21 or higher Requirement: Windows 2000, 2003, XP with Service Packs installed
2.0	03/07/2007		6.4		Added Client Components information
4.0	10/10/2009		1, 2, 3, 4, 6, 11		EHR/VueCentric Integration, BMX Monitor APPCONTEXTS
4.0 Patch 5	08/28/2019	Ken Halfpenny	11 (and review all other sections)		Update SAC (Standards and Conventions) Requirements section and general update to use the 2018 Document Template format.
4.0 Patch 5	11/18/2019	Ken Halfpenny	All		Update document based on comments from SAC review.

Preface

The purpose of this manual is to provide technical information about the *BMXNet ADO.NET Data Access and Connectivity Utilities for RPMS* (BMX) package. The BMX package is designed to support connectivity and data exchange between .NET applications and the Resource and Patient Management System (RPMS).

1.0 Introduction

BMXNet 4.0 represents a refactoring of BMXNet 2.0 to support Electronic Health Record (EHR)/VueCentric components, multisession connections, new dialogs, and a variety of enhancements.

This manual provides Indian Health Service (IHS) site managers with a technical description of the BMXNet routines, files, menus, cross references, globals, and other necessary information required to effectively manage the system.

BMX Client Connection Management is introduced in Version 4.0 along with APPCONTEXTS security feature, and minor bug fixes (see the BMXNet ADO.NET Data Access and Connectivity Utilities for RPMS User Manual for additional information).

All routines, files, options, and keys are namespaced starting with the letters “BMX.”

The file number range for this package is 90093.1–90093.99.

2.0 Orientation

See the Installation Instructions section in the Installation Guide for maintenance and implementation information.

3.0 Implementation and Maintenance

3.1 System Requirements

- VA FileMan version 22 or higher
- VA Kernel version 8.0 or higher
- Cache version 5.0.21 or higher
- Windows 2000, 2003, XP with Service Packs installed
- Microsoft .NET Framework v2.0.50727 or higher
- EHR/VueCentric 1.1 patch 5 for BMXEHR40.net
- IHS ICD/CPT LOOKUP & GROUPER (AICD) v4.0 for ICD10

3.2 Package-Wide Variables

There are no package-wide variables in the BMXNet package.

3.3 Security Keys

Table 3-1 Security key name and description

Key Name	Description
BMXZMENU	All BMXNet managers must have this security key.

4.0 Routine Descriptions

4.1 Routines with Description

Table 4-1 details routines and their associated descriptions. There are a total of 60 routines.

Table 4-1: Routines and descriptions

Routine	Description
BMXADE1	IHS/OIT/HMW–BMXNet ADO.NET PROVIDER ;
BMXADE2	IHS/OIT/HMW–BMXNet ADO.NET PROVIDER ;
BMXADO	IHS/CIHA/GIS–RPC CALL: GENERATE AN ADO SCHEMA STRING AND DATA SET
BMXADO2	IHS/CIHA/GIS–BMX ADO RECORDSET UTILS
BMXADOF	IHS/CIHA/GIS–RPC CALL FOR EXTENDED FUNCTIONALITY OF BMXNet UTILITIES
BMXADOF1	IHS/CIHA/GIS–RPC CALL FOR EXTENDED FUNCTIONALITY OF BMXNet UTILITIES
BMXADOF2	IHS/CIHA/GIS–RPC CALL FOR EXTENDED FUNCTIONALITY OF BMXNet UTILITIES
BMXADOFD	IHS/CIHA/GIS–RPC CALL FOR EXTENDED FUNCTIONALITY OF BMXNet UTILITIES
BMXADOF5	IHS/CIHA/GIS–RPC CALL FOR EXTENDED FUNCTIONALITY OF BMXNet UTILITIES
BMXADOI	IHS/CIHA/GIS–RPC CALL: GENERATE DATA FOR AN ADO DATASET
BMXADOS	IHS/CIHA/GIS–UPDATE THE BMX ADO SCHEMA FILE
BMXADOS1	IHS/CIHA/GIS–UPDATE THE BMX ADO SCHEMA FILE GUI VERSION
BMXADOV	IHS/CIHA/GIS–RPC CALL: GENERATE DATA FOR AN ADO DATASET
BMXADOV1	IHS/CIHA/GIS–RPC CALL: GENERATE DATA FOR AN ADO DATASET
BMXADOV2	IHS/CIHA/GIS–RPC CALL: GENERATE DATA FOR AN ADO DATASET
BMXADOVJ	IHS/CIHA/GIS–RPC CALL: GENERATE DATA FOR AN ADO DATASET
BMXADOX	IHS/CIHA/GIS - RPC CALL: GENERATE AN ADO SCHEMA STRING AND DATA SET
BMXADOX1	IHS/CIHA/GIS - RPC CALL: GENERATE AN ADO SCHEMA STRING AND DATA SET
BMXADOX2	IHS/CIHA/GIS - RPC CALL: GENERATE AN ADO SCHEMA STRING AND DATA SET
BMXADOXX	IHS/CIHA/GIS - RPC CALL: GENERATE AN ADO SCHEMA STRING AND DATA SET
BMXADOXY	IHS/CIHA/GIS - RPC CALL: GENERATE AN ADO SCHEMA STRING AND DATA SET
BMXE01	IHS/OIT/FJE–ENVIRONMENT CHECK FOR BMX 2.0 ;

Routine	Description
BMXEHR	IHS/VANGENT/GS – BMX PROTOCOL INTEGRATION FOR EHR/VUECENTRIC
BMXFIND	IHS/OIT/HMW–BMX GENERIC FIND
BMXG	IHS/OIT/HMW–UTIL: GET DATA
BMXGETS	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXMBRK	IHS/OIT/HMW–BMXNet MONITOR
BMXMBRK2	IHS/OIT/HMW–BMXNet MONITOR
BMXMEVN	IHS/OIT/HMW–BMXNet MONITOR
BMXMON	IHS/OIT/HMW–BMXNet MONITOR
BMXMSEC	IHS/OIT/HMW–BMXNet MONITOR
BMXNTEG	INTEGRITY CHECKER;OCT 31, 2006
BMXPO	IHS/CM/MAW–Populate AppContext with all Namespaced RPCs
BMXPRS	IHS/OIT/HMW–BMX WINDOWS UTILS
BMXRPC	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXRPC1	IHS/OIT/HMW–UTIL: REMOTE PROCEDURE CALLS
BMXRPC2	IHS/OIT/HMW–FIELD LIST
BMXRPC3	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXRPC4	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXRPC5	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXRPC6	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXRPC7	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXRPC8	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXRPC9	IHS/OIT/HMW–RPC CALL FOR EXTENDED BROKER FUNCTIONALITY
BMXRPC10	IHS/ VANGENT/GS –RPC CALL FOR EXTENDED BROKER FUNCTIONALITY
BMXSQL	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXSQL1	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXSQL2	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXSQL3	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXSQL4	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXSQL5	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXSQL6	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXSQL7	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXSQL91	IHS/OIT/HMW–BMX REMOTE PROCEDURE CALLS
BMXTABLE	IHS/OIT/HMW–BMX RETURN ENTIRE TABLE
BMXTRS	IHS/OIT/HMW–UPPERCASE-LOWERCASE
BMXUTL1	IHS/OIT/HMW–UTIL: PATIENT DEMOGRAPHICS
BMXUTL2	IHS/OIT/HMW–UTIL: PATIENT INFO

Routine	Description
BMXUTL5	IHS/OIT/HMW-DATE FORMAT
BMXUTL6	IHS/OIT/HMW-BMXNET INSTALLATION CALLS

5.0 Files and Tables

The following tables include information regarding file lists and file access.

5.1 File List

Table 5-1 details file numbers and names.

Table 5-1: File list information

File Number	File Name
90093.1	BMX USER
90093.2	BMX APPLICATION
90093.5	BMXNET MONITOR
90093.9	BMX GUI REPORT
90093.98	BMX ADO LOG
90093.99	BMX ADO SCHEMA

5.2 File Access

Table 5-2 lists file access information.

Table 5-2: File access information

FILE (#)	FILENAME	GL	RD	WR	LYG	DD	DEL
90093.1	BMX USER	^BMXUSER(@	@	@	@	@
90093.2	BMX APPLICATION	^BMXAPPL(@	@	@	@	@
90093.5	BMXNET MONITOR	^BMXMON(@	@	@	@	@
90093.9	BMX GUI REPORT	^BMXGUIR(#	#	#	@	#
90093.98	BMX ADO LOG	^BMXADOL(@	@	@	@	@
90093.99	BMX ADO SCHEMA	^BMXADO(@	@	@	@	@

5.3 Cross References

The BMX files all use the standard “B” cross reference as defined by FileMan. One additional cross reference is used in the BMX GUI REPORT file.

The PACKAGE field has a MUMPS cross reference to identify the package specifics of the report. The AUSRP cross reference identifies the cross reference as:

^BMXGUIR(“AUSRP,” User Pointer, Package Pointer, Inverse Start Date/Time, Record IEN). The following example shows how cross references may display on your computer screen.

```

FILE: BMX USER
GLOBAL: ^BMXUSER(
FILE #: 90093.1

```

FIELD #	FIELD NAME	SUBSCRIPT	PIECE	TYPE
.01	WINIDENT X: B 1) S ^BMXUSER("B", \$E(X,1,30), DA)=""	D0,0	1	F
.02	USER	"	2	P
.03	V ENCRYPTED	"	3	F

```

FILE: BMX APPLICATION
GLOBAL: ^BMXAPPL(
FILE #: 90093.2

```

FIELD #	FIELD NAME	SUBSCRIPT	PIECE	TYPE
.01	MAJOR VERSION X: B 1) S ^BMXAPPL("B", \$E(X,1,30), DA)=""	D0,0	1	F
.02	MINOR VERSION	"	2	F
.03	BUILD	"	3	D

```

FILE: BMXNET MONITOR
GLOBAL: ^BMXMON(
FILE #: 90093.5

```

FIELD #	FIELD NAME	SUBSCRIPT	PIECE	TYPE
.01	PORT X: B 1) S ^BMXMON("B", \$E(X,1,30), DA)=""	D0,0	1	N
.02	ENABLED	"	2	S
.03	INTEGRATED SECURITY	"	3	S
.04	SESSION NAMESPACE	"	4	F

```

FILE: BMX GUI REPORT
GLOBAL: ^BMXGUIR(
FILE #: 90093.9

```

FIELD #	FIELD NAME	SUBSCRIPT	PIECE	TYPE
.01	NAME X: B 1) S ^BMXGUIR("B", \$E(X,1,30), DA)=""	D0,0	1	F
.02	USER WHO REQUESTED REPORT	"	2	P
.03	START TIME	"	3	D
.04	END TIME	"	4	D
.05	OUTPUT TYPE	"	5	S
.06	OPTION NAME	"	6	F
.07	REPORT STATUS	"	7	S
.08	PACKAGE X: AUSRP^MUMPS 1) S ^BMXGUIR("AUSRP", \$P(^BMXGUIR(DA,0), "^", 2), X, (9999999.99999-\$P(^BMXGUIR(DA,0), "^", 3)), DA)=""	"	8	P

```

                2) K ^BMXGUIR("AUSRP", $P(^BMXGUIR(DA,0),"^",2),X,(9999999.
9999-$P(^BMXGUIR(DA,0),"^",3)),DA)

1100          OUTPUT (90093.911)
.01           OUTPUT                                D0,11,D1,0                1    W

                FILE: BMX ADO LOG
                GLOBAL: ^BMXADOL(
                FILE #: 90093.98

FIELD #       FIELD NAME                          SUBSCRIPT                          PIECE TYPE
=====
.01           TRANSACTION TIMESTAMP                D0,0                            1    D
                X: B
                1) S ^BMXADOL("B", $E(X,1,30),DA)=""
.02           FILE NUMBER                          "                                2    F
.03           DAS                                  "                                3    F

1             DATA (90093.981)
.01           DATA                                D0,1,D1,0                1    W

FILE: BMX ADO SCHEMA
                GLOBAL: ^BMXADO(
                FILE #: 90093.99

FIELD #       FIELD NAME                          SUBSCRIPT                          PIECE TYPE
=====
.01           SCHEMA NAME                          D0,0                            1    F
                X: B
                1) S ^BMXADO("B", $E(X,1,30),DA)=""
.02           FILE OR SUBFILE NUMBER                "                                2    N
.03           DATASET IS READ ONLY                  "                                3    S

1             FIELD NUMBER (90093.991)
.01           FIELD NUMBER                          D0,1,D1,0                1    F
                X: B
                1) S ^BMXADO(DA(1),1,"B", $E(X,1,30),DA)=""
.02           DATA TYPE                            "                                2    S
.03           FIELD LENGTH                          "                                3    F
.04           COLUMN HEADER                          "                                4    F
.05           READ ONLY                              "                                5    S
.06           KEY FIELD                              "                                6    S
.07           NULL ALLOWED                          "                                7    S
.08           IEN AUTOMATICALLY INCLUDED            "                                8    S
.09           ALWAYS GET INTERNAL VALUE              "                                9    S
1             AUTO IDENTIFIER EXTR FUNCT            D0,1,D1,1                E1,240  F
2             SPECIAL UPDATE EP                      D0,1,D1,2                E1,245  F
3             EXTR FUNCT FOR TRIGGERED VALU         D0,1,D1,3                E1,245  F
                E

2             VIEW (90093.992)
.01           VIEW                                  D0,2,D1,0                1    F
                X: B
                1) S ^BMXADO(DA(1),2,"B", $E(X,1,30),DA)=""
1             ENTRY POINT                            D0,2,D1,1                E1,240  F

2             PARAMETER (90093.9922)
.01           PARAMETER                              D0,2,D1,2,D2,0            1    F
                X: B
                1) S ^BMXADO(DA(2),2,DA(1),2,"B", $E(X,1,30),DA)=""

```

.02	BRIEF DESCRIPTION	"	2	F
3	DESCRIPTION (90093.9923)			
.01	DESCRIPTION	D0,2,D1,3,D2,0	1	W

Figure 5-1: Sample of cross-reference screens

5.4 Table File

Please see Figure 5-2 for example screens of a Table File.

```

FILE: BMX USER
GLOBAL: ^BMXUSER(
FILE #: 90093.1

```

FIELD #	FIELD NAME	SUBSCRIPT	PIECE	TYPE
.01	WINIDENT	D0,0	1	F
.02	USER	"	2	P
.03	V ENCRYPTED	"	3	F

```

FILE: BMX APPLICATION
GLOBAL: ^BMXAPPL(
FILE #: 90093.2

```

FIELD #	FIELD NAME	SUBSCRIPT	PIECE	TYPE
.01	MAJOR VERSION	D0,0	1	F
.02	MINOR VERSION	"	2	F
.03	BUILD	"	3	D

```

FILE: BMXNET MONITOR
GLOBAL: ^BMXMON(
FILE #: 90093.5

```

FIELD #	FIELD NAME	SUBSCRIPT	PIECE	TYPE
.01	PORT	D0,0	1	N
.02	ENABLED	"	2	S
.03	INTEGRATED SECURITY	"	3	S
.04	SESSION NAMESPACE	"	4	F
1	APPCONTEXTS (90093.51)			
.01	APPCONTEXT	D0,1,D1,0	1	P
2	DESCRIPTION	D0,2	E1,245	F

```

FILE: BMX GUI REPORT FILE
GLOBAL: ^BMXGUIR(
FILE #: 90093.9

```

FIELD #	FIELD NAME	SUBSCRIPT	PIECE	TYPE
.01	NAME	D0,0	1	F
.02	USER WHO REQUESTED REPORT	"	2	P
.03	START TIME	"	3	D
.04	END TIME	"	4	D

```

.05      OUTPUT TYPE           "           5      S
.06      OPTION NAME          "           6      D
.07      REPORT STATUS        "           7      S
.08      PACKAGE              "           8      P
1100     OUTPUT (90093.911)
  .01      OUTPUT              D0,11       1      W

          FILE: BMX ADO LOG FILE
          GLOBAL: ^BMXADOL(
          FILE #: 90093.98

FIELD #   FIELD NAME                SUBSCRIPT                PIECE TYPE
=====
.01      TRANSACTION TIMESTAMP      D0,0                    1      D
.02      FILE NUMBER                "                        2      F
.03      DAS                        "                        3      F
1        DATA (90093.981)
  .01      DATA                    D0,1                    1      W

          FILE: BMX ADO SCHEMA
          GLOBAL: ^BMXADO(
          FILE #: 90093.99

FIELD #   FIELD NAME                SUBSCRIPT                PIECE TYPE
=====
.01      SCHEMA NAME                D0,0                    1      F
.02      FILE OR SUBFILE NUMBER      "                        2      N
.03      DATASET IS READ ONLY        "                        3      S

1        FIELD NUMBER (90093.991)
  .01      FIELD NUMBER              D0,1,D1,0              1      F
  .02      DATA TYPE                "                        2      S
  .03      FIELD LENGTH              "                        3      F
  .04      COLUMN HEADER              "                        4      F
  .05      READ ONLY                  "                        5      S
  .06      KEY FIELD                  "                        6      S
  .07      NULL ALLOWED               "                        7      S
  .08      IEN AUTOMATICALLY INCLUDED "                        8      S
  .09      ALWAYS GET INTERNAL VALUE  "                        9      S
  .1      SECONDARY SCHEMA            "                       10     P
1        AUTO IDENTIFIER EXTR FUNCT  D0,1,D1,1              E1,240 F
2        SPECIAL UPDATE EP           D0,1,D1,2              E1,245 F
3        EXTR FUNCT FOR TRIGGERED VALU D0,1,D1,3              E1,245 F
        E

2        VIEW (90093.992)
  .01      VIEW                      D0,2,D1,0              1      F
1        ENTRY POINT                 D0,2,D1,1              E1,240 F

2        PARAMETER (90093.9922)
  .01      PARAMETER                 D0,2,D1,2,D2,0         1      F
  .02      BRIEF DESCRIPTION         "                        2      F

3        DESCRIPTION (90093.9923)
  .01      DESCRIPTION               D0,2,D1,3,D2,0         1      W
    
```

Figure 5-2: Sample of table file screens

5.5 Client Assembly Information

Three run-time assemblies are furnished with this application: the core BMXNET40.dll file and framework library files for either the Windows Client (BMXWIN40.dll) or the EHR/VueCentric (BMXEHR40.dll) lib directory. The attributes can be found by highlighting the assemblies and right-clicking on the file. The following information represents the released version of the assembly and can be found under the Version tab of the Properties menu.

Table 5-3: BMXNET40.dll version and assembly information

Assembly Version	4.0.0.2
Comments	ADO.NET 2.0/Context Core Library for RPMS
Company	Indian Health Service (IHS)
File Version	4.0.0.2
Internal Name	BMXNET40.dll
Language	Language Neutral
Original File Name	BMXNET40.dll
Product Version	4.0.0.2

Table 5-4: BMXWIN40.dll version and assembly information

Assembly Version	4.0.0.2
Comments	WinForm framework for BMX-based applications
Company	Indian Health Service (IHS)
File Version	4.0.0.2
Internal Name	BMXWIN40.dll
Language	Language Neutral
Original File Name	BMXWIN40.dll
Product Version	4.0.0.2

Table 5-5: BMXEHR40.dll version and assembly information

Assembly Version	4.0.0.2
Comments	EHR/VueCentric single-sign on adaptor for BMX
Company	Indian Health Service (IHS)
File Version	4.0.0.2
Internal Name	BMXEHR40.dll
Language	Language Neutral
Original File Name	BMXEHR40.dll
Product Version	4.0.0.2

6.0 External Relations

6.1 Callable Routines

Table 6-1: Callable routines and descriptions

Routine Called	Description
JEP^BMXADO	RECURSION RE-ENTRY POINT FOR JOINS
SS^BMXADO	RETURN THE SCHEMA STRING IN AN ARRAY
FILE^BMXADOF	RPC CALL: UNIVERSAL FILEMAN RECORD UPDATER UTILITY
BAFM^BMXADOF1	RPC: PASS DATA FROM A STD BROKER ADO ARRAY TO FILEMAN AND RETURN THE ACK MSG IN 'OUT' ARRAY
\$\$NARR^BMXADOF2	GET IEN OF PROVIDER NARR & UPDATE DATA STG FOR PROBLEM FILE
\$\$VVAR^BMXADOF2	CHECK SPECIAL VARIABLES REQUIRED FOR UPDATING THE VISIT FILE
\$\$FACNIEN^BMXADOFD	GIVEN A PROBLEM IEN AND FACILITY IEN, RETURN THE FACILITY-NOTE IEN
\$\$SPEC^BMXADOF5	SPECIAL DATA MODS FOR SPECIFIC FILES
\$\$FDEF^BMXADOS	GIVEN A FILEMAN FILE AND FIELD, RETURN THE DATA DEFINITION IN ADO FORMAT
FLIST^BMXADOS	GIVEN A FILE RETURN THE FILEDS IN AN ARRAY MODE=0: NUMERIC ORDER, MODE=1: ALPHA ORDER
SC^BMXADOS	SUB CRAWLER. GIVEN A FILE NUMBER RETURN ALL OF ITS DESCENDANT FILES IN AN ARRAY
\$\$IENS^BMXADOV	CONVERT DAS STRING TO IENS STRING
\$\$IXFLD^BMXADOV	GIVEN AN FILE NUMMER AND INDEX NAME, RETURN THE FIELD NUMBER, TYPE, AND DINUM SUBTYPE
VIEW^BMXADOV	VIEW A DATA SET ; GATEWAY TO ALL ITERATORS
\$\$AA^BMXADOV1	VISIT/V-FILE ITERATION USING THE 'AA' INDEX
\$\$AAP^BMXADOV1	ITERATOR FOR PROBLEM FILE: AA INDEX
\$\$LOOK^BMXADOV1	ITERATE BY A SINGLE STANDARD INDEX THAT IS A POINTER VALUE
\$\$LOOK1^BMXADOV1	ITERATE USING A STANDARD INDEX
\$\$LOOK2^BMXADOV1	TEXT POINTER LOOKUP
\$\$NUMIT^BMXADOV1	ITERATE BY NUMBER
DATA^BMXADOV1	ADD DATA NODES TO ARRAY
\$\$MCDIEN^BMXADOV2	GIVEN A PATIENT IEN, RETRUN THE IEN OF THAT PT'S MOST RECENT RECORD IN MEDICAID ELIGIBILITY FILE
JFLD^BMXADOVJ	STUFF JOIN FIELD IDS INTO THE INTRO SEGMENT OF THE SCHEMA

Routine Called	Description
JOIN^BMXADOVJ	APPEND ADDITIONAL ANRS TO FULFILL JOIN REQUESTS
DISP^BMXADOX	TEMP DISPLAY
FIND^BMXFIND	PLACE MATCHING RECORDS FROM REQUESTED FILE INTO A RESULT GLOBAL
\$\$GET^BMXG	RETURN TEXT OF .01 FIELD OF AN ENTRY IN A FILE
\$\$GETV^BMXMBRK	GET VALUE OF V - REFERENCE PARAMETER
\$\$PRSA^BMXMBRK	PARSE API INFORMATION, GET CALLING INFO
\$\$PRSB^BMXMBRK	PARSE PARAMETER
\$\$PRSM^BMXMBRK	PARSE MESSAGE
\$\$PRSP^BMXMBRK	PARSE PROTOCOL
CALLP^BMXMBRK	MAKE API CALL USING PROTOCOL STRING
LINST^BMXMBRK	INSTANTIATE LOCAL ARRAY
\$\$CREF^BMXMBRK2	CONVERT ARRAY CONTAINED IN P TO REFERENCE A
\$\$OARY^BMXMBRK2	CREATE STORAGE ARRAY
CAPI^BMXMBRK2	MAKE API CALL
UNREGALL^BMXMEVN	UNREGISTER ALL EVENTS FOR CURRENT SESSION
\$\$SEMAPHOR^BMXMON	LOCK/UNLOCK BMXMON SEMAPHORE
ADDCTXT^BMXMSEC	SUPPORTS DIRECT MODE IN BMX
CHKPRMIT^BMXMSEC	CHECKS TO SEE IF REMOTE PROCEDURE IS PERMITTED TO RUN
SET^BMXMSEC	SET THE STATE VARIABLE
SETUP^BMXMSEC	SETS UP ENVIRONMENT FOR GUI SIGNON
PARSE^BMXPRS	PARSE SQL STATEMENT INTO ARRAY
POST2^BMXPRS	REMOVE COMMAS FROM BMXTK
CVC^BMXRPC10	RPC: BMX CVC ; CHECK VERIFY CODE (SEE CVC^XUSRB)
\$\$WINUGET^BMXRPC3	RETURNS DUZ FOR USER HAVING WINDOWS IDENTITY BMXWINID
COLTYPE^BMXSQL	APPEND COLUMN TYPES AND WIDTH TO OUTPUT GLOBAL
KW^BMXSQL1	IDENTIFY AND MARK KEYWORDS IN BMXTK
SCREEN^BMXSQL1	STE UP BMXFG() ARRAY OF EXECUTABLE SCREEN CODE
\$\$DECSTR^BMXSQL2	DECREMENTS STRING COLLATION VALUE BY 1
\$\$FLDFILE^BMXSQL2	RETURNS NAME OF FILE CONTAINING FIELD BMXIN
\$\$INCSTR^BMXSQL2	INCREMENTS STRING COLLATION VALUE BY 1
SETX^BMXSQL2	SET UP EXECUTABLE SCREEN CODE BY ASSEMBLING PIECES IN BMXFG AND ATTACH TO EXECUTABLE ITERATOR(S)
BLDIT^BMXSQL3	BUIOLD ITERATOR

Routine Called	Description
PLEVEL^BMXSQL3	ANALYZE WHERE STATEMENT ACCORDING TO PAREN LEVEL
XRTST^BMXSQL3	RETURNS TRUE 1) in BMXRET IF 'NORMAL' INDEX EXISTS FOR FIELD IN BMXFF(BMXNDX)
JOIN^BMXSQL4	JOIN PROCESSING
SELECT^BMXSQL5	GET FIELD NAMES INTO BMXFLD("NAME")="FILE#^FIELD#"
\$\$CHKCR^BMXSQL7	RETURNS CROSS REFERENCE TO ITERATE ON FOR RELATED FILE
WHERE^BMXSQL7	WHERE CLAUSE PROCESSING
\$\$T^BMXTRS	TRANSLATE WORD TO MIXED CASE
\$\$AGEF^BMXUTL1	RETURN AGE FORMATTED "35 Months" or "23 Years"
\$\$DOB^BMXUTL1	RETURN PATIENT'S DATE OF BIRTH IN FILEMAN FORMAT
\$\$HRCN^BMXUTL1	RETURN IHS HEALTH RECORD NUMBER
\$\$SEXW^BMXUTL1	RETURN PATIENT SEX: "Female"/"Male".
\$\$NEXTAPPT^BMXUTL2	RETRUN PATIENT'S NEXT APPOINTMENT FROM SCHEDULING PACKAGE
\$\$TXDT1^BMXUTL5	RETURN EXTERNAL DATE IN FORMAT:
DD-MMM-YYYY@HH:MM	
FROM FILEMAN FORMAT (internal):	
YYMMDD.HHMM	

6.2 Published Entry Points

Table 6-2: Published entry point routine and description

Routine	Description
EVENT^BMXMEVN	<p>Raise event to interested clients. Clients are listed in ^TMP("BMX EVENT",BMXEVENT,BMXSESS)=DUZ Input Parameters:</p> <ul style="list-style-type: none"> • BMXEVENT – the event • BMXPARAM – additional parameter list • BMXORIG – the event originator's session • BMXKEY – a ~-delimited list of security keys

6.3 Exported Options

Please see Table 6-3 for option names and descriptions.

Table 6-3: Exported options information

Option Name	Description
BMX MONITOR EDIT	Use this option to add/edit any BMXNet monitors in the BMXNET MONITOR file.
BMX MONITOR START	Use this option to start or restart all BMXNet monitors in the BMXNET MONITOR file. This option should be scheduled as a STARTUP type option in TaskMan. Do not use this option to start a specific monitor. To do this, in programmer mode, do STRT^BMXMON(PORT). See the product documentation for instructions on how to start session monitors in a particular namespace and on how to enable or disable Windows Integrated Security.
BMX MONITOR STOP	Use this option to stop all BMXNet monitors in the BMXNET MONITOR file.
BMX MONITOR VIEW	Use this option to view all BMXNet monitors in the BMXNET MONITOR file.
BMXMENU	Menu contains the above options in the BMX namespace
BMXRPC	All BMXNet users must have access to this option. This option contains a list of the basic RPCs required to use BMXNet.

6.4 Client Components

The BMXNET BMXNET40.dll, BMXWIN40.dll, and BMXEHR40.dll libraries are exported by the package for the development in applications. Assemblies are targeted to .NET Framework 2.0 for the development of WinForm client applications or EHR/VueCentric components. The BMXNET40.dll has been obfuscated to protect sensitive source code. The object is written in C# using Visual Studio 2008 IDE. There are no additional dependencies. There are no third-party controls used to create the BMXNET assemblies.

Visual Studio 2008 IDE

- Microsoft Visual Studio 2008 Version 8.0.50727.762 (SP.050727-7600)

BMXNET40.dll, BMXEHR40.dll, BMXWIN40.dll

- Machine Intel 386
- Characteristics
 - The file is a .NET Assembly
 - Line numbers stripped from file
 - Local symbols stripped from file
- Operating System 32-bit/64-bit Windows XP or later
- .NET Framework v2.0 or later
- File Type: Dynamic Linked Library (dll)

7.0 Internal Relations

There are no internal relations in the BMXNet package.

8.0 Archiving and Purging

No archiving or purging is necessary with BMXNet.

9.0 Generating Online Documentation

This section describes a few methods to generate online technical documentation.

9.1 System Documentation

Online BMXNet system documentation can be generated through the use of several Kernel options, including, but not limited to:

- %INDEX
- Menu Management
- Inquire Option
- Print Option File
- VA FileMan
- Data Dictionary Utilities
- List File Attributes

For more option listings and further information about other utilities that supply online technical information, see the Decentralized Hospital Computer Program (DHCP) Kernel Reference manual.

9.1.1 %INDEX

This option analyzes the structure of a routine to determine in part, if the routine adheres to RPMS programming standards. The output can include the following components:

- Compiled list of errors and warnings
- Routine listing
- Local variables
- Global variables
- Naked globals
- Label references
- External references

Running %INDEX for a specified set of routines allows users to discover any deviations from RPMS programming standards that exist in the selected routines and to see how routines interact with one another (i.e., which routines call or are called by other routines).

To run %INDEX for the BMXNet package, type the BMX namespace at the “Routine(s)?” prompt.

9.1.2 Inquire Option

The Inquire menu management option provides the following information about a specified option:

- Option name
- Menu text
- Option description
- Type of option
- Lock (if any)

In addition, all items on the menu are listed for each menu option. To secure information about BMXNet options, specify the BMX namespace.

9.1.3 Print Option File

The Print Option File utility generates a listing of options from the Option file (#19). Users can print all of the entries or a single option or range of options.

9.1.4 List File Attributes

This VA FileMan option allows users to generate documentation pertaining to files and file structure. The standard format of this option provides the following data dictionary information for a specified file:

- File name and description
- Identifiers
- Cross-references
- Files pointed to by the file specified
- Files that point to the file specified
- Input, print, and sort templates

In addition, the following applicable data is supplied for each field in the file:

- Field name, number, title, and description
- Global location
- Help prompt
- Cross-references
- Input transform
- Date last edited
- Notes

Using the Global Map format of this option generates an output that lists the following information:

- All cross-references for the file selected
- Global location of each field in the file
- Input, print, and sort templates

For a comprehensive listing of BMXNet files, see Section 5.0.

9.1.5 Online Help

In addition to system documentation, RPMS includes special help displays for most menu options and data entry prompts. Type a question mark (?) at the “Select . . . Option” prompt to display information related to the current option, as shown in Table 9-1.

Table 9-1: Online help input and display

Typing . . .	Displays . . .
one question mark (?)	a list of all options accessible from the current option.
two question marks (??)	a list of all accessible options and their formal names
three question marks (???)	a brief description for each option in a menu.
one question mark (?) followed by an option name (? OPTION)	extended help, if available, for that option

10.0 SAC Requirements and Exemptions

Applicable standards are listed in Table 10-1.

Table 10-1: Standards and descriptions

Standard Number	Description
2.2.3.1.4	SET DUZ
2.2.3.2.5.1	Use of ^TMP(\$J
2.2.3.3.2	Intrinsic Variables direct use is prohibited
2.2.4.3	CLOSE, direct use prohibited
2.2.4.4	HALT command
2.2.4.5	JOB, direct use prohibited
2.2.4.9	OPEN, is prohibited

Reasons for Exemption

BMXNet is a system level utility that mediates connections between .NET applications and the RPMS environment. As such, BMXNet requires the ability to directly open and close TCP/IP socket connections, access intrinsic variables, manipulate DUZ and its descendants, and directly invoke the JOB and HALT commands. The ^TMP(is used without \$J subscript to temporarily store data to support asynchronous RPC calls that span jobs.

SAC Exemption Approval list

Table 10-2 lists specific code updates that have been approved for a SAC exemption in addition to the generic approvals listed above.

Table 10-2: SAC exemption approvals

Patch	Program	Line	Exception	Date Approved
BMXv4.0p5	BMXMON	CONNERR+7	Use of \$ZI	7/3/2019
BMXv4.0p5	BMXMON	+153	Use of \$ZA	7/3/2019

Glossary

ADO.NET

Set of classes that expose data access services to the .NET programmer.

Application Program Interface

Callable entry points that enable software to communicate with other software.

AppContext

See Application Context.

Application Context

A read/write part of application programs that contains rules dictating how the application interacts with other applications and users. Application Context is implemented in RPMS as an option.

BMXNet

A set of software utilities designed to connect to RPMS data by .NET applications.

EHR/VueCentric

Single sign-on application used by IHS as the primary EHR. The VueCentric framework is the underlining COM-based technology that supports component deployment, layout, communications, and management.

column_alias

Specifies an alternative name to replace the column name in the query result set. For example, an alias such as “Quantity,” “Quantity to Date,” or “Qty” can be specified for a column named “quantity.”

field_name

Specifies from which field (column) the FROM clause should return. The field_name is associated with the preceding table_name by using a dot (.) between the two.

FROM Keyword

Specifies that the system should return the values from the following table_list statement.

INDEX Keyword

Identifies a specific FileMan cross-reference to use when retrieving data. Always use the SHOWPLAN keyword in conjunction with the INDEX keyword to ensure that the intended result is achieved.

Index Row

Part of the SHOWPLAN results that includes the M code created by BMXNet that will execute on the RPMS server and iterate through the FileMan file.

Iterate

To say or perform again; repeat.

join_type

Specifies a join using nonstandard syntax and the WHERE clause. The =* operator is used to specify a one-to-many (OTM) join. Use the OTM join to express relationship between Tables A and B such that a record in Table A can be referenced by a FileMan pointer field in one or more records in Table B.

Log on

Connect to a network.

M

A programming language that originated from the medical sector but is currently used in a variety of database applications due to its retrieval capabilities.

M Routines

A collection of command lines, all associated with a single name that can be stored and retrieved as a unit.

MAXRECORDS Keyword

Specifies the maximum number of records to return.

Operator

A symbol that specifies which operation the systems should perform relative to the indicated operator arguments.

Operator Arguments

A value or expression dictating the information upon which the related operator acts.

Overloads

The creation of more than one procedure, instance constructor, or property in a class with the same name but different argument types. Overloading is especially useful when your object model dictates that you employ identical names for procedures that operate on different data types.

Parameter

A value given to a variable until the related operation is completed. Parameters are treated by the system as constants. Parameters are often used to customize a program for a particular purpose.

Port

Software that links one computer with another using TCP/IP address and port numbers assigned by network administrators.

primary_table_name

The file from which the system should first retrieve data.

related_table_name

The file from which the system should receive data that matches the primary table.

Remote Procedure Call (RPC)

A technique used to construct distributed, client-server based applications. RPC extends the capabilities of local procedure calling so that the called procedure need not exist in the same address space as the calling procedure.

Screen Rows

Part of the SHOWPLAN results that includes the M code that BMXNet will execute to filter the results. The caret (^) character in the query plan is replaced by the tilde (~) character.

search_condition

Restricts the rows returned in the result set by using predicates. There is no limit to the number of predicates that can be included in a search condition.

SELECT Keyword

Specifies that the system should return the values by the following select_list statement.

select_list

Specifies which fields (columns) the system should select for the returned set. The select_list a series of expressions separated by commas.

Server

A computer that hosts RPMS applications.

SHOWPLAN Keyword

Returns the query plan, including the M code that will be executed to retrieve the records.

table_name

Specifies from which files (tables) the FROM clause should return fields.

table_source

Specifies from which files (tables) the FROM clause should return fields.

WHERE Keyword

Specifies a search condition to restrict the rows returned.

Acronym List

Acronym	Meaning
DHCP	Decentralized Hospital Computer Program
dll	Dynamic Linked Library
EHR	Electronic Health Record
IHS	Indian Health Service
RPC	Remote Procedure Call
RPMS	Resource and Patient Management System
SAC	Standards and Conventions
VA	Department of Veterans Affairs

Contact Information

If you have any questions or comments regarding this distribution, please contact the IHS IT Service Desk.

Phone: (888) 830-7280 (toll free)

Web: <http://www.ihs.gov/itsupport/>

Email: itsupport@ihs.gov