



BFX Migration Tool
for
Unisys OS2200 Systems

Release 1.0.4

User Guide

Revision Record

Revision	Description
01 (09/2018)	Initial release.
1.0.1 (6/2019)	Clarify starting the appropriate BFXJS; Fixes the following issues: 8000 Program path must allow for BFX program name 8004 Increase file size limit to 26. Force an ending . Unisys only 8317 Add dbank support for migtool 8339 Non-alphanumeric characters allowed for version 8340 Does not parse 1.2 BFX parameters 8341 MIG tool does not appear to follow parser rules for comments 8342 Unisys offers file truncated after 256 characters 8348 Note BFXJS startup proc changes required when replacing current code path with MIG tool
1.0.2 (8/2019)	8388 MIG301E Errors when making successive calls to Migtool 8408 Add a #NoOffersFound for BFX jobs without an ID= (JOBSUBMIT)
1.0.3 (8/2020)	8797 Manual installation does not reference changing #NoOffersFound
1.0.4 (11/2020)	8920 Maximum offers limited to 300 8930 Issue tracker Unisys migtool 1.0.3 8935 Offer matching issue 8937 MIG313E not issued for 3 character code versions 8939 MIGXXXC messages are inconsistent 8941 Update message appendix

© 2020 by Network Executive Software. Reproduction is prohibited without prior permission of Network Executive Software. Printed in the U.S.A. All rights reserved.

The U.S. Department of Commerce restricts the distribution of technical information contained in this document when exported outside the U.S. Therefore, careful attention should be given to compliance with all applicable U.S. Export Laws if any part of this document is to be exported.

Comments regarding this document may be submitted by addressing e-mail to:

support@netex.com

or, by visiting our web site at:

<http://www.netex.com>

Always include the complete title of the document with your comments.

Preface

This manual describes the BFX Migration Tool software for Netex customers deploying new versions of BFX products on the Unisys OS2200 platforms. The migration tool consists of a front end that allows a customer to choose which processes use one of several BFX versions/products to facilitate a smooth transition.

This manual is intended for all users of the migration tool and contains all of the information necessary to expand the user's ability to the fullest extent of the software.

The manual is divided into 3 parts:

“BFX Migration Tool Overview” gives a basic description of the migration utility.

“Installation Procedures” explains the installation steps by operation system.

“Using the Migration Tool” describes how to use the Migration Tool.

Notice to the Reader

This product is intended for use only as described in this document. Network Executive Software cannot be held responsible for use of features or options in any undocumented manner. This manual is subject to change without notice.

This document and the programs described in it are furnished under a license from Network Executive Software and may be used, copied, and disclosed only in accordance with such license.

This document contains references to the trademarks of the following corporations.

Corporation Trademarks and Products

Network Executive Software	NetEx, NetEx/IP
International Business Machines	IBM, zOS, AIX
Unisys	OS2200

These references are made for informational purposes only.

The diagnostic tools and programs described in this manual are **not** part of the products described.

Document Conventions

The following notational conventions are used in this document.

Format	Description
displayed information	Information displayed on a CRT (or printed) is shown in this font.
<i>user entry</i>	<i>This font</i> is used to indicate the information to be entered by the user.
UPPERCASE	The exact form of a keyword that is not case-sensitive or is issued in uppercase.
MIXedcase	The exact form of a keyword that is not case-sensitive or is issued in uppercase, with the minimum spelling shown in uppercase.
bold	The exact form of a keyword that is case-sensitive and all or part of it must be issued in lowercase.
lowercase	A user-supplied name or string.
value	Underlined parameters or options are defaults.
<label>	The label of a key appearing on a keyboard. If “label” is in uppercase, it matches the label on the key (for example: <ENTER>). If “label” is in lowercase, it describes the label on the key (for example: <up-arrow>).
<key1><key2>	Two keys to be pressed simultaneously.
No delimiter	Required keyword/parameter.

Definitions

BFX	Bulk File Transfer is a family of NetEx software products. Secure BFX is an equivalent but incompatible product family that supplies security to the transfers.
NetEx™	NETWORK EXECUTIVE software; a proprietary transport protocol.

Contents

Revision Record	ii
Preface.....	iv
Notice to the Reader.....	vi
Corporation Trademarks and Products.....	vi
Document Conventions.....	vii
Definitions.....	vii
Contents	ix
Figures.....	ix
BFX Migration Tool Overview	1
Introduction.....	1
BFXJS	2
System Requirements.....	2
OFFER Table	2
Installation Procedures.....	5
Unisys OS2200	5
Using the Migration Tool	7
BFXJS	7
Inserting the Migration Tool into the Production Execution Path	7
Migrating to a New BFX Version.....	7
Appendix A. Migration Tool Messages	9

Figures

Figure 1. Sample OFFER Table File	1
Figure 2. Visual overview of BFX Migration Tool implementation.	2

BFX Migration Tool Overview

Introduction

The BFX Migration is a utility which runs in tandem with one or more versions of BFX or Secure BFX on a platform. The purpose of this utility is to allow the customer to select which BFX jobs will run with various versions of installed BFX/Secure BFX. However, Secure BFX jobs can only run between hosts that run Secure BFX and BFX jobs can only run between hosts that run BFX (legacy). This tool allows customers to migrate to newer versions of BFX or Secure BFX based on BFX Jobs.

In the past, customers typically tested a new version, and when they were ready, cut the new version of software into production. With the release of Secure BFX, this conversion would require all systems using BFX to cutover at the same time. If one system had a problem, all systems would have been required to fall back to the previous version. This tool will allow customers to migrate from one release of BFX to a new release based on the BFX OFFERID. Fall back procedures would consist of removing the OFFERID rules except the default rule which points to the initial version on each system.

The migration tool utilizes a customizable file to specify which BFX version jobs use based on OFFERID. If no offer ID is found, the entry in the NoOffersFound is utilized. This should only be the case when a JOBSUBMIT command is executed, without a SEND or RECEIVE. The file utilizes some wildcard characters to ease in migrations. Once all Jobs are migrated to the newer BFX version the old version can be removed, and the customer can continue to run as is. When another version is installed, this tool can be used to migrate to the new version by installing the new version and adding rules to the OFFER Table to systematically test the new version selectively by job.

This tool is based on a text-based OFFER TABLE. Depending on the OFFER name, different versions of BFX can be executed. If the run stream contains multiple Offer IDs, the last OFFER Id will determine the code load. This is an example of an OFFER table file.

```
#ProgramFiles
# Maximum code loads 3
# V1 is the UDP version of BFX
V1 H301*RELEASE147.
# V2 is the Extend mode UDP BFX
V2 H301E*RELEASE344.
# V3 is the secure BFX
V3 H305*RELEASE110.
#
#OfferNames
# Maximum offers 3000
tes* V2
te%t* V3
* V1
#NoOffersFound
* V1
```

Figure 1. Sample OFFER Table File

BFXJS

The migration tool does not impact the BFXJS. You must start a copy of BFXJS for each BFX version you will be using. If you are running a H301 version of BFX, you must start a BFXJS from the H301 library if you wish to submit jobs. If you run the H305 version of BFX, you must start a BFXJS from the H305 library if you wish to submit jobs. If you wish to use H305 to run both secure job submission and non-secure job submission, you must start two copies of BFXJS, one running secure, one running nonsecure.

System Requirements

Installation of each BFX version must be placed in a unique library. Both local and remote hosts must run compatible products (i.e. Secure BFX H305 on a Unisys OS2200 host to Secure BFX H215 on an IBM zOS host).

BFX MIGRATION TOOL OVERVIEW

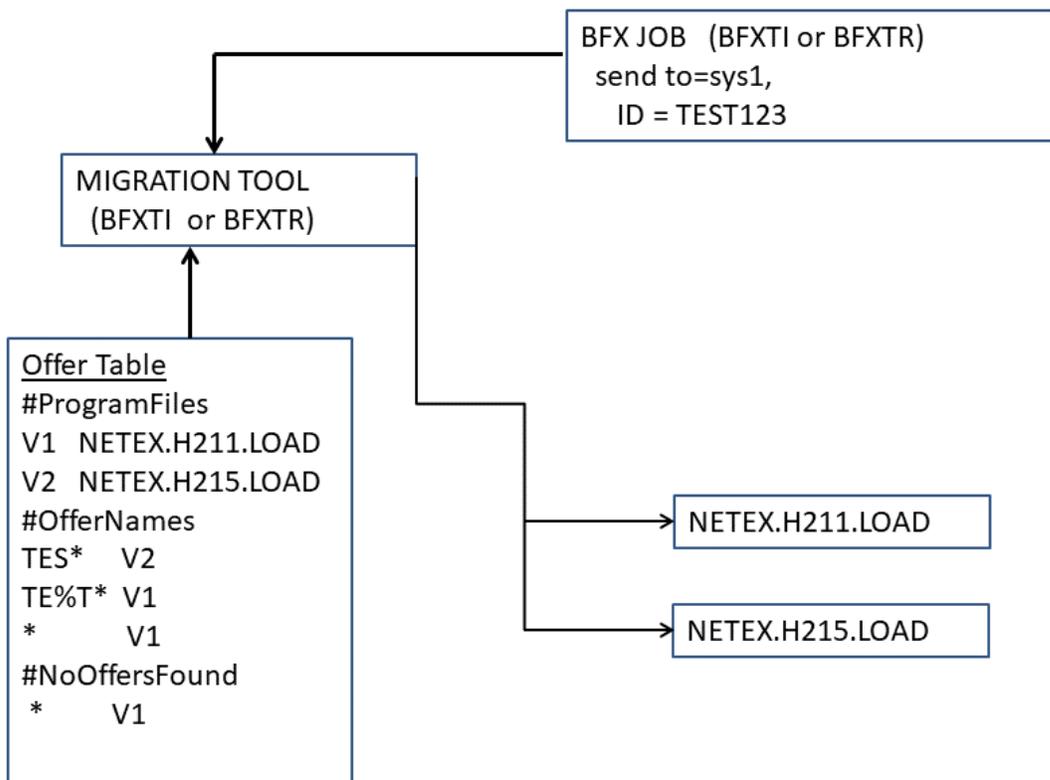


Figure 2. Visual overview of BFX Migration Tool implementation.

OFFER Table

The OFFER Table file contains fixed name tags and values which allow the site to customize the migration of new versions of the BFX or Secure BFX on a JOB basis. The OFFER Table file will be located in the library

the migtool is running from as the member name OFFERS. This member must exist. It must contain at least a default statement to utilize one of the versions of BFX or Secure BFX. This is a text file. Comments may be added to this file by coding a hash mark (#) in the first column, anywhere after the #ProgramFiles statement. The following describes the keywords and wildcard characters.

Fixed Name Tag	Value	Description
#ProgramFiles	<i>NONE</i>	Must be the first line in this file. This identifies the ProgramFiles section of the OFFER File.
#		Comment. Must be in the first column. Comments are ignored by the program but are helpful for describing the intentions of the OFFER file.
[version ID]	<i>[location of the version]</i>	A unique two-character identifier followed by at least one space and the location of the load library. Each ID must be unique (i.e. V1, V2). (Maximum of 3 entries.) Refer to Note 1.
#OfferNames	<i>NONE</i>	Defines the start of the BFX Job OFFER names (IDs).
[OFFERIDS]	<i>[versionID]</i>	Each entry in the OFFER name section is an OFFERID followed by at least one space and the two-character identifier referring to the version of code to execute. The OFFERID may contain the following wildcards representing any characters or string. (Maximum of 3000 entries) Refer to Note 2.
*	<i>[wildcard string]</i>	This character matches one or more characters to the end of the string in the comparison of an OFFERID.
%	<i>[wildcard character]</i>	This character matches exactly one character in the comparison of an OFFERID.
#NoOffersFound		This is a section Identifier. It is required.
*	<i>[versionID]</i>	When an OFFERID is not found in the BFX run stream, this entry will specify the version of code to be executed. Only 1 entry can be listed.

Note 1: All data may be entered in upper, lower or mixed cases alphanumeric characters (i.e. V1 is the same as v1). For systems that use lowercase names, they will remain as entered. All other systems will be converted to UPPERCASE.

Note 2: On all systems, OFFERIDs will be converted to UPPERCASE.

OFFERIDs are compared to the OFFERID in the BFX job. The order of the compare is from top to bottom. The wildcard characters are helpful shortcuts when groups of jobs are to be run on one version of the product. The first match will be the entry used.

Using the OFFER Table file in in Figure 1 on page 1, if the BFX job specified an OFFERID of TEST123, the V2 code load would be used. TEST123 matches TES* rule. If the OFFERID of TEXT123 was used, TEXT123 does not match the TES* rule. The next rule will be inspected. The next rule TE%T* matches TEXT123, so the V3 code base will be used. It is recommended that an entry specifying an * be the last rule in the table. This will match any offer name. In the sample table, the V1 code load would be the default. If a match is not found, the program ends in error.

When migrating to the Secure BFX, the tables on the local side and the remote side must match to ensure the connection can be established. The (legacy) UDP version of BFX cannot communicate with the Secure products. These tables are read and checked at the start of each job. Updating the table is effective immediately.

Installation Procedures

Unisys OS2200

NOTE: If you copy the BFXTR/BFXTI to TPF\$ before execution, you must also copy the OFFERS symbolic. Offers is located in the library BFXTI and BFXTR is executed from.

1. Contact NetEx Support to receive a download link.
2. Ftp this file to your Unisys system.
 - a. You must specify both “bin” and “quote site cfmt”
 - b. You may name this file using your dataset naming conventions
3. Update the OFFERS member in this library.
 - a. Update the #ProgramFiles sections with the code identifiers and the location of the corresponding BFX load library.
 - b. Update the #OfferNames section as required.
 - c. Your last entry should be an * to specify the default BFX load library to use.
 - d. Update the #NoOffersFound section as required.
4. Install the required Dbank support for H300Ipc
 - a. In H300ipc, users create additional executables for the various dbanks required. The Migration tool **MUST** have the same program names as all the executables used in the basic mode BFX code. These same names must also exist in the secure BFX load library. The Migration tool will always execute the same program name in the load library that was selected. These can be unique program names like BFXTI10, BFXTI20, BFXTR10, and BFXTR20 or they could be versioned program names like BFXTI/1101-1 and BFXTR/1101-1. If the name exists in the basic mode library, it must exist in the migration tool library and the secure bfx load library.

For the Migration product:

@copy,a MIGT*LOAD.BFXTI, MIGT*LOAD.BFXTI10

@copy,a MIGT*LOAD.BFXTR, MIGT*LOAD.BFXTR10

For the secure product:

@copy,a H305*BFXLOAD.BFXTI, H305*BFXLOAD.BFXTI10

@copy,a H305*BFXLOAD.BFXTR, H305*BFXLOAD.BFXTR10

Using the Migration Tool

You may point a bfx test job to the migration tool load library for initial testing. The migration tool load library contains both a BFXTI and a BFXTR executable. They both process the table in the same manner. It will list the contents of the Offer table, the OFFERID in the job, which rule was matched, and the load library used. If your job copies these members to tpf\$, you must also copy the offers member.

BFXJS

The migration tool does NOT impact or work with BFXJS in any way. You will still need to start the appropriate copies of BFXJS from each library to ensure proper operation. If the dataset name changes during the “install into production” phase, then the corresponding BFXJS run stream must be updated to point to the new location for BFXJS.

Inserting the Migration Tool into the Production Execution Path

The migration tool was designed to be implemented so a site can continue to operate production jobs while systematically testing a new version on a job by job basis. Initially, the Migration Tool will need to be inserted into the execution path for BFXTI and BFXTR. The System Administrator should perform the following steps:

1. Copy your current production BFX load library to a new dataset.
2. Update your BFXJS proc to point to the new dataset location.
3. Delete your current production BFX load library and reallocate it. Copy the migration tool (migtool) library, to this library.
4. Edit the OFFERS member in this file. Records in this file must be under the standard 256 characters used by the Unisys “C” compiler.
 - a. In the #ProgramFiles section, enter a descriptor and the new dataset name for your old production library. In the #OffersName section, ensure the * rule is specified and the descriptor for your new BFX load library.
5. Now all jobs should run as before.

Migrating to a New BFX Version

1. Install the new version of BFX.
2. In the ProgramFiles section, specify a descriptor and the dataset name for the new version (up to three load libraries can be tested.)
3. Edit the OFFER Table to add the OFFERID of the job to test and point it to the new version descriptor.
4. Ensure the * rule is the last rule in the file, and the associated code load to use.
5. If testing is being done to another system, ensure it is running compatible products (i.e. Secure BFX only runs to other Secure BFX products.)
6. Run the test job.

7. Repeat with more jobs as necessary.
8. When you are ready to make the new version the new production, change the default rule at the bottom to point to the new version descriptor. All specific OFFERID rules can be deleted.

NOTE: If at any time a back out is necessary, the rules for the specific OFFERIDs can be removed from all OFFER Table files and leave the default rule so that all jobs are running with the initial version of BFXTI/BFXTR.

Appendix A. Migration Tool Messages

The migration tool generates a variety of messages during execution. Shown below is a complete list of messages with the suggested response for each. Also shown is the severity of the message.

MIGnnns message text

- MIG** This indicates that this is a Migration Tool message.
- nnn** This is the error number. Messages are listed in this order.
- s** This indicates the message severity. The following codes are used:
- I** - informational messages
 - W** - warning messages
 - E** - error messages
 - C** -continuation message

message text This area displays the text of the message.

The following are the messages issued by the Migration Tool.

MIG100I BFX Migration Tool %s

Severity: I

Explanation: The migration tool is logging the start of an execution, and the release level.

User Response: This is informational only. No response is required

MIG100C %s

Severity: I

Explanation: Displays the copyright date

Response: This is informational only. No response is required

MIG101I Offer Table='%s'

Severity: I

Explanation: The migration tool is logging the location of the offers control table.

User Response: This is informational only. No response is required

MIG102I Using Offer ID='%s'

Severity: I

Explanation: The migration tool is logging the offer id used in the BFX job.

User Response: This is informational only. No response is required

MIG103I Match found Offer=%s, Table=%s

Severity: I

Explanation: The migration tool is logging the matching offer table entry.

User Response: This is informational only. No response is required

MIG104I Using Code %s**Severity:** I**Explanation:** The migration tool is logging the code version that will be used to locate the code base.**User Response:** This is informational only. No response is required**MIG105I Using Load Library %s****Severity:** I**Explanation:** The migration tool is logging the code location used for the file transfer.**User Response:** This is informational only. No response is required**MIG106I Found Offer=%s****Severity:** I**Explanation:** The migration tool is logging offer id found in the BFX job.**User Response:** This is informational only. No response is required**MIG107I Migration Table****Severity:** I**Explanation:** The migration tool is printing the contents of the offer table**User Response:** This is informational only. No response is required**MIG109I %s: line %d****Severity:** I**Explanation:** The migration tool is logging the offer control file name and line number associated with the last error message issued. The contents of the line may also be displayed in a continuation message.**User Response:** This is informational only. No response is required**MIG109C %s****Severity:** I**Explanation:** Displays additional information from the previous error.**User Response:** This is informational only. No response is required**MIG201W ID= was coded with no offerid****Severity:** W**Explanation:** The migration tool failed to find the offer id in the BFX parameters. The NoOffersFound entry will be used. The table will be used as if the offered was not found.**User Response:** Ensure the ID parameter is coded.**MIG202W Offer ID=%s is greater than 8 characters****Severity:** W**Explanation:** While processing the BFX input, an offer id was found. It was over the 8-character maximum. The NoOffersFound entry will be used. The table will be used as if the offered was not found.**User Response:** Correct the BFX input.

MIG301E Allocation of tpf.bfxtemp for 1000 failed

Severity: E

Explanation: The migration tool is attempting to create a file in TPF\$. The allocation failed.

User Response: The meaning of the error is printed. Correct the error or try increasing the allocation of TPF\$

MIG301C %s

Severity: E

Explanation: The meaning of the error is printed.

User Response: Correct the error or try increasing the allocation of TPF\$

MIG302E Open failed for bfxtemp in tpf\$

Severity: E

Explanation: The migration tool attempted to open a file in TPF\$. The reason for the failure is listed.

User Response: Correct the error and rerun the job.

MIG302C %s

Severity: E

Explanation: The meaning of the error is printed.

User Response: Correct the error.

MIG306E No match found

Severity: E

Explanation: The migration tool did not match any entries in the offer table.

User Response: Add the offer id to the table or ensure the last entry is an *. This will match any offer.

MIG307E Code version %snot found

Severity: E

Explanation: The code version specified in the offer table does not match any listed in the ProgramFiles section.

User Response: Ensure all offers match a valid code version in #ProgramFiles.

MIG308E Code Location must end in a “.”

Severity: E

Explanation: The code location displayed does not end with a “.”.

User Response: Add the ending “.” to the code location.

MIG309E Line exceeds maximum size of %lu

MIG309E %s

Severity: E

Explanation: The migration tool read a line that exceeded the maximum.

User Response: Break the line into smaller lines.

MIG310E #ProgramFiles must be the first statement

MIG310E %s

Severity: E

Explanation: The offer table file does not start with #ProgramFiles.

User Response: Ensure the #ProgramFiles statement is the first line in the offer file.

MIG311E Code location exceeds %i characters

Severity: E

Explanation: The path to the BFX code exceeds the maximum.

User Response: Correct the input, shorten the path, or contact technical support.

MIG312E Maximum of 3 code loads maybe specified

Severity: E

Explanation: More than three code versions were defined in the #ProgramFiles section of the offer table file.

User Response: Delete unused entries in the #ProgramFile section.

MIG313E Code versions are limited to 2 characters

Severity: E

Explanation: Code versions are limited to two characters.

User Response: Correct the code identifier. If this is an offer id, the #OfferNames card is missing at the end of the #ProgramFiles section and before the input for the offers.

MIG314E #OfferNames not found -- required

Severity: E

Explanation: While processing the offer table the #OfferNames statement was not found. This statement is required after the #ProgramFiles and the associated entries.

User Response: Insert this statement before any offer entries in the offer table.

MIG316E Offer patterns are limited to 8 characters

Severity: E

Explanation: The offer pattern exceeds the 8-character limit.

User Response: Correct the offer pattern.

MIG317E Maximum of 3000 offer patterns may be specified

Severity: E

Explanation: More than 3000 offer patterns were entered into the offer table.

User Response: Use of wildcarding should allow the entries to be consolidated. A % will match a single character. An * will match to the end of the offer.

MIG323E Can't open offer table %s errno=%i

Severity: E

Explanation: The migration tool failed to open the offer table file. The reason for the failure is displayed.

User Response: Ensure the correct file is being used, and the file exists with read permissions.

MIG324E Allocation of addfile for 1000 failed

MIG324C %s

Severity: E

Explanation: The migration tool attempting to assign a file in TPF and failed. The reason for the failure is displayed.

User Response: Correct the error.

MIG325E Open failed for addFile in tpf\$

MIG325C %s

Severity: E

Explanation: The migration tool failed to open a file in TPF\$. The reason is displayed.

User Response: Correct the error.

MIG326E Open failed for bfxlcl input in tpf\$

MIG326C %s

Severity: E

Explanation: The migration tool attempting to read in the BFX parameters from TPF\$. The reason for the failure is displayed. .

User Response: Correct the error.

MIG327E FACSf of @add failed rc=%i

MIG327C %s

Severity: E

Explanation: When attempting to @add in the updated run, FACSf issued a return code The reason is displayed.

User Response: Correct the error.

MIG330E UCSINFO func 3 failed status=%i

Severity: E

Explanation: An error occurred while issuing func code 3 for UCSINFO.

User Response: Check the UCSINFO documentation for the meaning of the error. Correct and rerun.

MIG331E UCSINFO func 17 failed status2=%i

Severity: E

Explanation: An error occurred while issuing func code 17 for UCSINFO.

User Response: Check the UCSINFO documentation for the meaning of the error. Correct and rerun.

MIG332E Code version not specified

Severity: E

Explanation: A 2-character code version was missing in the #OfferNames or #NoOfferFound section. **User Response:** Correct the entry in the offer table file.

MIG334E Code location not specified

Severity: E

Explanation: While processing the #ProgramFiles section, a code version was entered, without specifying the code location.

User Response: Add the code location for this code version.

MIG335E Duplicate code version %s

Severity: E

Explanation: While processing the #ProgramFiles section a duplicate code version was found. The value is printed.

User Response: Make the code descriptor unique or delete the duplicate.

MIG337E UCSINFO func 13 status=%i

Severity: E

Explanation: An error occurred while issuing func code 13 for UCSINFO.

User Response: Check the UCSINFO documentation for the meaning of the error. Correct and rerun.

MIG340E Only 1 entry is allowed in NoOffersFound

Severity: E

Explanation: Only 1 entry is allowed in the NoOffersFound section.

User Response: Delete the extra entries.

MIG341E #NoOffersFound not found -- required"

Severity: E

Explanation: The #NoOffersFound section is missing. This is required.

User Response: Add a line to the offers table: #NoOffersFound

The next line should be an * followed by at least one space and the version of code to be executed when an Offer ID does not exist.

MIG342E #NoOffersFound requires 1 entry

Severity: E

Explanation: Only 1 entry is allowed, and it must be coded

User Response: Specify an * followed by the version of code you wish to execute when no Offer ID is specified.

MIG343E #OfferNames requires at least 1 entry

Severity: E

Explanation: At least 1 entry is required in this section.

User Response: The last entry should be * *version* to be used when a match was not made from previous entries. The version specifies the code load to use.

MIG344E #ProgramFiles requires at least 1 entry

Severity: E

Explanation: At least 1 entry must be specified. The maximum is 3 entries.

User Response: This entry contains a 2 character version followed by the dataset name of the BFX code. The version id is used in the other sections of the table

MIG345E Offer name must be * in the #NoOffersFound section

Severity: E

Explanation: An * is the only valid offer name in the NoOffersFound section.

User Response: Change the offer id to an *