IBM Financial Transaction Manager for SWIFT Services for z/OS 3.2.4

Readme
Fix Pack 10

This edition applies to Version 3 Release 2.4 of IBM Financial Transaction Manager for SWIFT Services for z/OS (5655-FTB) - Fix Pack 10 (3.2.4.10).

Reference key: 20230831-1051

Contents

General information.....5
Download location..... 5
Prerequisites and co-requisites.....5
What's new in FTM SWIFT..... 6
What's new in FTM SWIFT Version 3 Release 2.4 Fix Pack 10..... 6
What's new in FTM SWIFT Version 3 Release 2.4 Fix Pack 9.....7
What's new in FTM SWIFT Version 3 Release 2.4 Fix Pack 8.....7
Known Problems..... 8
Installation information.....9
Installing FTM SWIFT 3.2.4.10 – Create a new installation..... 9
Installing FTM SWIFT 3.2.4.10 – Update an existing installation..... 9
Separated file systems: Preparing and Switching..... 10
Shared file system: Preparing and Switching..... 13
Cleaning up..... 16
Falling back to the previous fix pack level..... 17
Re-migrating after a previous fallback.....19
Maintenance tasks.....21
Ensure that no customization operation is pending..... 21
Ensure that no configuration or security administration change is pending.....21
Use IBM Installation Manager to install the fix pack..... 22
Install a fix pack using wizard mode.....22
Install a fix pack using command line mode..... 23
Permissions of the installed files..... 23
Sharing files with the customization and runtime systems..... 24
Setting up the use of shared library regions.....24
Granting access permissions to FTM SWIFT users..... 24
Update customization definition data, and create deployment instructions and vehicles..... 25
Prepare BAR files for manual deployment.....26
Stop all FTM SWIFT related message flows.....26
Verifying the installation of the database routines..... 27
Deploy BAR files.....27
Re-activate FTM SWIFT accounting..... 28
Restart all FTM SWIFT related message flows..... 28
Recover the customization system.....28
Roll back the IBM Installation Manager update of the fix pack..... 29
Roll back using wizard mode.....29
Roll back using command line mode..... 29
Update an SAG Add-On..... 30
Prepare the migration of configuration entities..... 30
Migrate the configuration entities..... 31
Providing Db2 administration modules.....32
Saving configuration and security data..... 34
Restoring configuration and security data..... 34
Update the IBM Integration Toolkit workstation..... 35
Migrating FTM SWIFT table spaces to universal table spaces (UTS)..... 37
Copyright and trademark information.....49
Document change history.....51

General information

Before starting with the installation process, view the online version of this readme file to check if information has changed since the readme file was downloaded.

Download location

You can download FTM SWIFT 3.2.4.10 from Fix Central:

https://www.ibm.com/support/fixcentral/

Search for the Fix ID **3.2.4-FTM-SWS-ZOS-fp0010**.

Prerequisites and co-requisites

Before installing the current fix pack, perform the following steps:

- Check the hardware and software requirements of the fix pack you plan to install:
  Go to https://www.ibm.com/support/docview.wss?uid=swg27027034
  and select version **V3.2** and product **FTM for SWIFT Services for z/OS**.

  Updates of pre-requisite software must not be performed during fix pack installation and migration. It is a separate activity:

  - If your software is not at the minimum version required by the new fix pack, upgrade it to a level supported by your current installation and the new fix pack before you start the fix pack installation and migration activity.
  - If the new fix pack provides support for a new software version, install this new version only after you finished the installation and migration activity of the fix pack.
- Review the the Financial Transaction Manager support web site:
  https://www.ibm.com/support/pages/node/6346924
- Ensure that you have at least 500 MB of free disk space to contain the uncompressed installation image.
- If you already have FTM SWIFT installed:
  - If you have obtained special fixes, contact IBM Support to determine whether you need an updated version of the fixes before you install this fix pack.
  - Ensure that you have at least fix pack 3.2.4.7 installed and all post-installation steps were finished.

What's new in FTM SWIFT

The following sections summarize what has changed in updates of FTM SWIFT since fix pack 7 (3.2.4.7).

For a list of fixes provided and APARs included in the various product updates refer to: https://www.ibm.com/support/pages/node/6242258

Attention:

It is assumed that the migration to universal table spaces as described in "Migrating FTM SWIFT table spaces to universal table spaces (UTS)" on page 37 is completed before installation of Version 3 Release 2.4 Fix Pack 10. However, Version 3 Release 2.4 Fix Pack 10 does not rely on having this migration to be fully completed, but any later fix pack might require so.

What's new in FTM SWIFT Version 3 Release 2.4 Fix Pack 10

Update of the installed SAG Add-On required:	☐ Yes	☑ No

The following changes were introduced:

Support for SWIFT Standards Release 2023 (SR2023) added

This includes:

- FIN:
  - Message standards as specified in *Message Format Validation Rules - November 2023 Standards Release*
  - Validation of ISN messages for the mandatory SWIFT gpi services gCCT, gCOV, and gSRP as documented in *SWIFT gpi - Supplementary Message Format Validation Rules, 23 May 2022*.
  - MTXML schema files for FIN SR2023
- FINplus:
  - The DNIFINPLUS message domain is enhanced to provide the finplus2023 message definition set, providing the CBPR+ 2023 message types, the already existing CBPR+ 2.1 types, and the FINplus Securities SR2023 messages types. The FINplus Securities message type DRAFT5semt.044.001.01 that was available in the finplus2022 message definition set is not provided anymore.
- MX

The provided standards release update activates automatically according to the release schedule of SWIFT. However, you can install this fix pack before the SR2023 live date and continue to process messages according to the current standards release. If you want to use the new standards release updates for testing purposes earlier, you can activate them by using FTM SWIFT configuration. For details, see Testing a new message definition set.

Lists of all supported message types per message domain are provided in directory *DNIvINST*`/run/doc`. The format of the file names is *DOMAIN*`.messages.types` (for example, `DNIFINPLUS.messages.types`).

Removed support for SWIFT Standards

The support for the following SWIFT Standards is removed:

- FIN (SR2020)
- MX (SR2020)
- FINplus (SR2020)

Security enhancements for Web Applications

Several enhancements are included in FTM SWIFT enterprise applications to further improve the security.

Message Entry and Repair (MER):

Enhanced Copy/Paste operation for multi-line edit fields [FTMSWIFT-I-148]
Multi-line edit fields now allow pasted data to be auto-formatted according to the row and column settings of the field.

Remote address lookup for messages in the DNIFINPLUS domain [FTMSWIFT-I-94]

MER now supports remote address (distinguished name) lookup and expansion using BIC information in the remote address field. A new field which shows the current SWIFT service name of the selected transfer option set is added to the formatted view.

Reference Data Utility (RDU) supports SWIFTRef ReachPlus for FINplus [FTMSWIFT-I-137]

RDU is able to import SWIFTRef ReachPlus for FINplus files containing data related to the SWIFTNet FINplus service.

What's new in FTM SWIFT Version 3 Release 2.4 Fix Pack 9

The following changes were introduced:

Expand AO Bank Data Application to manage SWIFTNet remote addresses
The Administration and Operation (AO) web application can now manage SWIFTNet remote addresses for a bank data record.

RM filtering configurable
RM filtering is now configurable and can be made optional. RM filtering done by SWIFT is not affected.

New code signing certificate
A new code signing certificate is provided because a new code signing service provider is used by IBM.

Note: You need to import this certificate into your keystore as part of the migration procedure as described in this readme file.

Security enhancements for Web Applications
Several enhancements are included in FTM SWIFT enterprise applications to further improve the security.

What's new in FTM SWIFT Version 3 Release 2.4 Fix Pack 8

The following changes were introduced:

Message Entry and Repair (MER):

Remote address validation for messages in the DNIFINPLUS domain
MER now supports validation of the remote address field according to the SWIFTNet naming and addressing guidelines.

Number format configurable for DNIFINPLUS messages [FTMSWIFT-I-122]
Message Entry and Repair enterprise application supports number formatting for DNIFINPLUS messages, including browser print. You can configure thousands and decimal separators of numbers, for example for amounts.

BIC expansion for fields within messages of the DNIFINPLUS domain [FTMSWIFT-I-124]
Lookup and field expansion of a DNIFINPLUS message which contains BIC information are now supported in message entry and in browser printing. This change excludes the remote address information.

Additional attribute extraction for messages in the DNIFINPLUS domain [FTMSWIFT-I-131]

The values for Reference, Amount and Date are now extracted and displayed for the message types:

- pacs010
- camt029

Find message extended [FTMSWIFT-I-126]
The search criterion for finding messages by reference has been extended to allow dot (".") characters.

Error information for UNPARSABLE messages are now shown during edit
The error information which cause a message be marked as unparsable are now shown in the Unparsable edit dialog.

MSIF
MSIF will no longer reject incoming Store and Forward (SnF) messages from SWIFT if they fail the RM authorization check. Instead, applications will receive a MsgReceived notification with completion code `Partial0k` and reason codes DNFL9430E or DNFL9425I.

The Broker Administration Program (BAP) has been enhanced to support IIB and later versions.
The Broker Administration Program (BAP) has been enhanced to support IIB and later versions. This change includes updated program outputs and changes to error messages issued by BAP.

Updated administration modules to support UTS table spaces
Db2 administration modules have been changed to include information for UTS table spaces.

Message printing service

Number format configurable for DNIFINPLUS messages [FTMSWIFT-I-122]
Printing supports number formatting for DNIFINPLUS messages. You can configure thousands and decimal separators of numbers (for example, amounts) for the message printing service.

BIC expansion for fields within messages of the DNIFINPLUS domain [FTMSWIFT-I-124]
BIC information of messages in the DNIFINPLUS message domain are now expanded if enabled. This change excludes the remote address information.

Print layout changes:

- Line length information is considered when calculating header field offsets.
- Unparsable messages/incorrect header information is marked in printout using delimiter lines (v----v----...-----v-----v).

Timestamp format of history entries configurable [FTMSWIFT-I-141]
The configuration in which format timestamps are to be printed by the message printing service is now also used for history entries.

Security enhancements for Web Applications
Several enhancements are included in FTM SWIFT enterprise applications to further improve the security to prevent of Denial of Service and authorization vulnerability attacks.

New code signing certificate
A new code signing certificate is provided with extended validity period.

Note: You need to import this certificate into your keystore as part of the migration procedure as described in this readme file.

Known Problems

For a list of known problems refer to:
https://www.ibm.com/support/pages/node/6242088

Installation information

You can find information about the installation and migration steps mentioned in this document in the IBM Documentation for FTM SWIFT for z/OS:

http://www.ibm.com/docs/en/ftmswsfz324

This readme document uses the following variables:

Variable	Description	Default
inst_dir	The installation directory of FTM SWIFT.	/usr/lpp/IBM/ftm/ftmswift/v324
run_dir	The directory for runtime data.	/var/ftmswift_v324/run
cust_dir	The directory for customization data.	/var/ftmswift_v324/cus
deployment_dir	The deployment data directory.	/var/ftmswift_v324/cus/depdata
instance	The name of the FTM SWIFT instance.	INST1
ou	The name of the organizational unit.	Depending on the context this might be SYSOU, DNFSYSOU, or the name of a business OU
admin_ds_prefix	The data set prefix for Db2® administration modules.	
db2_ssid	The ID of the Db2 subsystem.	
applenv	The name of the Workload Manager (WLM) application environment.	

Installing FTM SWIFT 3.2.4.10 – Create a new installation

If you have not yet installed FTM SWIFT, follow the description in the IBM Documentation for FTM SWIFT to install and customize a new instance instead of using this readme file.

Installing FTM SWIFT 3.2.4.10 – Update an existing installation

Updating an existing environment consists of the phases *Preparing*, *Switching*, *Cleaning up* and optionally *Falling back*.

Depending on how you share your product files, there are two installation variants that differ in the amount of migration steps you can prepare before entering the downtime during which you cannot process workload:

Separated file systems
The file systems of the installation system and the customization/runtime systems are separated. The fix pack installation only affects the installation system until you manually share the files with your customization and runtime system. This helps you to prepare migration steps while your system can still process workload.

Shared file system
Your installation, customization and runtime environment use a single shared file system. The fix pack installation may immediately affect your runtime environment. This reduces the steps you can do to prepare the migration while your system can still process workload.

Choose the subsection that applies to your file system setup.

Separated file systems: Preparing and Switching

Follow the steps required to prepare and switch your environment.

Preparing

Perform the following steps while your runtime system continues to process workload:

1. If you use MER, identify outdated messages and templates with the MER message administration utility using the *migratelist* command.
   For templates, perform a migration action on the detected templates using the MER message administration utility *migtpl* command. Outdated messages should be archived, printed or exported.
2. If you use general purpose routing or MER routing message flows, check the impact of SR2023 changes to your routing logic and prepare required modifications.
3. Ensure that no customization operation is pending.
4. Ensure that no configuration or security administration change is pending.
5. Create a backup of your customized administrative scripts from *deployment_dir*/*instance*/admin:

   ```
   mkdir ~/admin_scripts_backup
   cp /var/ftmswift_v324/cus/depdata/INST1/admin/* ~/admin_scripts_backup
   ```

6. Create a backup of the following instance administration data set *prefix.DNIvINST.ADMIN* members:
   - DNIMZCFS
   - DNIMZCFR
   - DNIMZCFO
   - DNIMZRST
   - DNIMZREO
   - DNFMZRBD
   - DNFMZREO
   - DNFMZRST
7. Use IBM Installation Manager to install the fix pack for FTM SWIFT 3.2.4.10.
8. Share the files in the *inst_dir*/admin directory with your customization system.
9. Update customization definition data, and create deployment instructions and vehicles.
10. If you plan manual deployment of the FTM SWIFT BAR files, follow Prepare BAR files for manual deployment.
11. Prepare the migration of configuration entities.
12. Save configuration and security data. Refer to Saving configuration and security data.
13. If your migration starting point is FTM SWIFT fix pack level 3.2.4.7 or 3.2.4.8:
    Backup your certificate keystore, for example:

    ```
    cp -p /var/ftmswift_v324/run/ftmswift_keystore.jks /your_backup_directory
    ```

Switching

Perform the following steps during a scheduled downtime:

1. Stop all sessions and services you use.
2. Stop all FTM SWIFT application servers.
3. Back up the FTM SWIFT IBM WebSphere Application Server (WAS) profiles.
4. Restart all FTM SWIFT application servers.
5. Stop all FTM SWIFT enterprise applications.

6. Stop all FTM SWIFT related message flows.
7. Stop all FTM SWIFT message brokers.
8. Share the files in the *inst_dir*/run directory with your runtime system.
9. If your migration starting point was FTM SWIFT fix pack level 3.2.4.7 or 3.2.4.8:
   Replace the Public Key certificate for signed Java components using the *keytool* program from a Java Developement Kit (JDK):
   a. Delete the previous version of the certificate from your keystore:

      ```
      keytool -delete -alias ftmswift -keystore ks.jks
      ```

      where **ks.jks** refers to your keystore, e.g. /var/ftmswift_v324/run/ftmswift_keystore.jks. When prompted, enter the password of your keystore.
   b. Import the new version of the certificate into your keystore:

      ```
      keytool -importcert -alias ftmswift -file FTMSWIFT.cer -keystore ks.jks
      ```

      where:

      ks.jks
      Your certificate keystore, e.g. /var/ftmswift_v324/run/ftmswift_keystore.jks

      FTMSWIFT.cer
      The Public Key certificate from your FTM SWIFT installation directory, e.g. /usr/lpp/IBM/ftm/ftmswift/v324/run/cert/FTMSWIFT.cer

      When prompted, enter the password of your keystore. Check if the displayed certificate information is identical with the one provided below. Especially, check if the certificate fingerprints match.

      ```
      Owner: CN=International Business Machines Corporation, OU=IBM CCSS, O=International Business
      Machines Corporation, L=Armonk, ST=New York, C=US
      Issuer: CN=DigiCert Trusted G4 Code Signing RSA4096 SHA384 2021 CA1, O="DigiCert, Inc.", C=US
      Serial number: 9ae51c9cfb701bc6f01688fab2acfc9
      Valid from: 1/10/23 1:00 AM until: 11/15/24 12:59 AM
      Certificate fingerprints:
               MD5:  4E:A5:C4:0B:73:EA:23:54:24:A8:44:0F:07:06:16:70
               SHA1: F6:DB:01:1C:A3:D9:F2:80:00:2A:29:56:52:4B:9E:43:F9:EE:EF:7A
               SHA256:
      85:F7:23:4A:9E:A6:38:7D:C8:90:16:CD:CF:21:80:98:02:92:E9:87:9F:04:22:47:9A:3A:5B:5F:4A:0E:DE:E0
               Signature algorithm name: SHA256withRSA
               Version: 3
      ```

      If the displayed information is correct, confirm that you trust this certificate to finish the import.
10. To verify that the certificate was replaced successfully, run the Software Integrity Checker (SIC):

    ```
    dnpsic -Djava.security.policy=/var/ftmswift_v324/run/ftmswift.policy
    ```

    The check needs to finish successfully.
11. Open and follow the deployment instructions.

    If your migration starting point is FTM SWIFT fix pack level 3.2.4.7 and if you deploy the deployment vehicle for resource class DB, the deployment vehicle may cause the following warning for an insert statement for the table *DNQE_ME_DNIFINPLUS*:

    ```
    IWAQ0003W SQL warnings were found
    SQLState=02000 ROW NOT FOUND FOR FETCH, UPDATE, OR DELETE, OR THE RESULT OF A QUERY IS AN
    EMPTY TABLE.
    ```

    You can safely ignore the IWAQ0003W warning. However, you must ensure that no other warnings or errors are shown.

    If you do not plan to use generated deployment vehicles for resource class CFGPF, you need to manually update the following enterprise applications:

Table 1. IBM WebSphere Application Server applications to be updated			
Application	**Migrating from FP7**	**Migrating from FP8**	**Migrating from FP9**
Administration and Operation (AO)	Yes	Yes	Yes
Message Entry and Repair (MER)	Yes	Yes	Yes
Relationship Management Application (RMA)	Yes	Yes	Yes
WebHome enterprise application	Yes	Yes	Yes

12. Update the Db2 administration modules by copying the following JCL files from the *deployment_dir*/*instance*/admin into the instance administration dataset *prefix.DNIvINST.ADMIN*
    - DNIMZCFS.JCL
    - DNIMZCFR.JCL
    - DNIMZCFO.JCL
    - DNIMZRST.JCL
    - DNIMZREO.JCL
    - DNFMZRBD.JCL
    - DNFMZREO.JCL
    - DNFMZRST.JCL

    Reapply any modifications to customize the administration modules to your environment.
13. Refresh your WLM application environment:

    ```
    V WLM,APPLENV=applenv,REFRESH
    ```
14. Follow the instruction in Verifying the installation of the database routines.
15. Restart all FTM SWIFT message brokers.
16. Deploy BAR files.
17. Verify the deployed BAR files:

    ```
    dniczbap -cmd list
    ```

    The deployment was successful if the displayed versions contain 3.2.4.10.
18. Re-activate FTM SWIFT accounting if you use the SIPN FIN or FMT FIN service.
19. Restart all FTM SWIFT related message flows.
20. Migrate the configuration entities.
21. Run the following script files if you have both MSIF and MER facility deployed:
    - dnqctos.cli
    - dnqctosrole.cli

    See Configuring transfer option set field for details.
22. Restart all FTM SWIFT enterprise applications.
23. It is recommended to set the property **InvalidateOnUnauthorizedSessionRequestException** in the IBM WebSphere Application Server. See step 5 in Configuring security settings and custom properties for the application server for details. Then restart the IBM WebSphere Application Server.
24. Restart all sessions and services.
25. Update the IBM Integration Toolkit workstation If you use either of the following:
    - FTM SWIFT sample message flows as foundation for your own flow development

- FTM SWIFT nodes in your own message flows
- FTM SWIFT message set projects containing XML schema definitions that, for example, are utilized by the IBM Integration Toolkit XPath wizard

Replace the file `com.ibm.dnq.api.jar` in the Toolkit `dropins` directory with the updated file from *`inst_dir`*`/admin/toolkit`.

Check the impact of SR2023 changes to your routing logic and prepare required modifications. Rebuild and redeploy all BAR files that contain message flows that use FTM SWIFT API nodes.

26. If you have both MSIF and MER facility deployed and want to expand BIC addresses using SWIFTRef ReachPlus for FINplus information, load the corresponding reference data using the Reference Data Utility. See Reference Data Utility command for details.
27. If you use Message Entry and Repair (MER) with messages of domain DNIFINPLUS, it is required to perform an additional step to identify outdated messages and templates with the MER message administration utility using the *migratelist* command. Perform a migration action on the detected templates using the MER message administration utility *migtpl* command.

    Note:

    Perform the commands using a special properties file provided for the sole purpose of fix pack migration. Use the `-properties` option when running the Administration Utility with *`inst_dir`*`/run/data/DnqFINPLUS_FP10.xml`.

```
dnpadm migtpl -outputfile finplus.csv -properties /usr/lpp/IBM/ftm/ftmswift/v324/run/data/
DnqFINPLUS_FP10.xml
```

After you finished the fix pack migration, you can continue to convert table spaces that you did not yet migrate to UTS. For details, see “Migrating FTM SWIFT table spaces to universal table spaces (UTS)” on page 37.

Shared file system: Preparing and Switching

Follow the steps required to prepare and switch your environment.

Preparing

Perform the following steps while your runtime system continues to process workload:

1. If you use MER, identify outdated messages and templates with the MER message administration utility using the *migratelist* command.
   For templates, perform a migration action on the detected templates using the MER message administration utility *migtpl* command. Outdated messages should be archived, printed or exported.
2. If you use general purpose routing or MER routing message flows, check the impact of SR2023 changes to your routing logic and prepare required modifications.
3. Ensure that no customization operation is pending.
4. Ensure that no configuration or security administration change is pending.
5. Create a backup of your customized administrative scripts from *`deployment_dir`*`/`*`instance`*`/admin`:

```
mkdir ~/admin_scripts_backup
cp /var/ftmswift_v324/cus/depdata/INST1/admin/* ~/admin_scripts_backup
```

6. Create a backup of the following instance administration data set *prefix.DNIvINST.ADMIN* members:
   - DNIMZCFS
   - DNIMZCFR
   - DNIMZCFO
   - DNIMZRST
   - DNIMZREO

- DNFMZRBD
- DNFMZREO
- DNFMZRST

7. Save configuration and security data. Refer to Saving configuration and security data.
8. If your migration starting point is FTM SWIFT fix pack level 3.2.4.7 or 3.2.4.8:
   Backup your certificate keystore, for example:

```
cp -p /var/ftmswift_v324/run/ftmswift_keystore.jks /your_backup_directory
```

Switching

Perform the following steps during a scheduled downtime:

1. Stop all sessions and services you use.
2. Stop all FTM SWIFT application servers.
3. Back up the FTM SWIFT IBM WebSphere Application Server (WAS) profiles.
4. Restart all FTM SWIFT application servers.
5. Stop all FTM SWIFT enterprise applications.
6. Stop all FTM SWIFT related message flows.
7. Stop all FTM SWIFT message brokers.
8. Use IBM Installation Manager to install the fix pack for FTM SWIFT 3.2.4.10.
9. If your migration starting point was FTM SWIFT fix pack level 3.2.4.7 or 3.2.4.8:
   Replace the Public Key certificate for signed Java components using the *keytool* program from a Java Developement Kit (JDK):

   a. Delete the previous version of the certificate from your keystore:

```
keytool -delete -alias ftmswift -keystore ks.jks
```

   where **ks.jks** refers to your keystore, e.g. `/var/ftmswift_v324/run/ftmswift_keystore.jks`. When prompted, enter the password of your keystore.

   b. Import the new version of the certificate into your keystore:

```
keytool -importcert -alias ftmswift -file FTMSWIFT.cer -keystore ks.jks
```

   where:

   ks.jks
   Your certificate keystore, e.g. `/var/ftmswift_v324/run/ftmswift_keystore.jks`

   FTMSWIFT.cer
   The Public Key certificate from your FTM SWIFT installation directory, e.g. `/usr/lpp/IBM/ftm/ftmswift/v324/run/cert/FTMSWIFT.cer`

   When prompted, enter the password of your keystore. Check if the displayed certificate information is identical with the one provided below. Especially, check if the certificate fingerprints match.

```
Owner: CN=International Business Machines Corporation, OU=IBM CCSS, O=International Business
Machines Corporation, L=Armonk, ST=New York, C=US
Issuer: CN=DigiCert Trusted G4 Code Signing RSA4096 SHA384 2021 CA1, O="DigiCert, Inc.", C=US
Serial number: 9ae51c9cfb701bc6f01688fab2acfc9
Valid from: 1/10/23 1:00 AM until: 11/15/24 12:59 AM
Certificate fingerprints:
         MD5:  4E:A5:C4:0B:73:EA:23:54:24:A8:44:0F:07:06:16:70
         SHA1: F6:DB:01:1C:A3:D9:F2:80:00:2A:29:56:52:4B:9E:43:F9:EE:EF:7A
         SHA256:
85:F7:23:4A:9E:A6:38:7D:C8:90:16:CD:CF:21:80:98:02:92:E9:87:9F:04:22:47:9A:3A:5B:5F:4A:0E:DE:E0
         Signature algorithm name: SHA256withRSA
         Version: 3
```

   If the displayed information is correct, confirm that you trust this certificate to finish the import.

10. To verify that the certificate was replaced successfully, run the Software Integrity Checker (SIC):

    ```
    dnpsic -Djava.security.policy=/var/ftmswift_v324/run/ftmswift.policy
    ```

    The check needs to finish successfully.
11. Update customization definition data, and create deployment instructions and vehicles.
12. Open and follow the deployment instructions.

    If your migration starting point is FTM SWIFT fix pack level 3.2.4.7 and if you deploy the deployment vehicle for resource class DB, the deployment vehicle may cause the following warning for an insert statement for the table *DNQE_ME_DNIFINPLUS*:

    ```
    IWAQ0003W SQL warnings were found
    SQLState=02000 ROW NOT FOUND FOR FETCH, UPDATE, OR DELETE, OR THE RESULT OF A QUERY IS AN
    EMPTY TABLE.
    ```

    You can safely ignore the IWAQ0003W warning. However, you must ensure that no other warnings or errors are shown.

    If you do not plan to use generated deployment vehicles for resource class CFGPF, you need to manually update the following enterprise applications:

    Table 2. IBM WebSphere Application Server applications to be updated

Application	**Migrating from FP7**	**Migrating from FP8**	**Migrating from FP9**
Administration and Operation (AO)	Yes	Yes	Yes
Message Entry and Repair (MER)	Yes	Yes	Yes
Relationship Management Application (RMA)	Yes	Yes	Yes
WebHome enterprise application	Yes	Yes	Yes

13. Update the Db2 administration modules by copying the following JCL files from the *deployment_dir*/*instance*/admin into the instance administration dataset *prefix.DNIvINST.ADMIN*

    - DNIMZCFS.JCL
    - DNIMZCFR.JCL
    - DNIMZCFO.JCL
    - DNIMZRST.JCL
    - DNIMZREO.JCL
    - DNFMZRBD.JCL
    - DNFMZREO.JCL
    - DNFMZRST.JCL

    Reapply any modifications to customize the administration modules to your environment.
14. Refresh your WLM application environment:

    ```
    V WLM,APPLENV=applenv,REFRESH
    ```

15. Follow the instruction in Verifying the installation of the database routines.
16. Restart all FTM SWIFT message brokers.
17. If you plan manual deployment of the FTM SWIFT BAR files, follow Prepare BAR files for manual deployment.
18. Deploy BAR files.

19. Verify the deployed BAR files:

    ```
    dniczbap -cmd list
    ```

    The deployment was successful if the displayed versions contain 3.2.4.10.
20. Re-activate FTM SWIFT accounting if you use the SIPN FIN or FMT FIN service.
21. Restart all FTM SWIFT related message flows.
22. Prepare the migration of configuration entities.
23. Migrate the configuration entities.
24. Run the following script files if you have both MSIF and MER facility deployed:
    - dnqctos.cli
    - dnqctosrole.cli

    See Configuring transfer option set field for details.
25. Restart all FTM SWIFT enterprise applications.
26. It is recommended to set the property **InvalidateOnUnauthorizedSessionRequestException** in the IBM WebSphere Application Server. See step 5 in Configuring security settings and custom properties for the application server for details. Then restart the IBM WebSphere Application Server.
27. Restart all sessions and services.
28. Update the IBM Integration Toolkit workstation If you use either of the following:
    - FTM SWIFT sample message flows as foundation for your own flow development
    - FTM SWIFT nodes in your own message flows
    - FTM SWIFT message set projects containing XML schema definitions that, for example, are utilized by the IBM Integration Toolkit XPath wizard

    Replace the file `com.ibm.dnq.api.jar` in the Toolkit `dropins` directory with the updated file from *`inst_dir`*`/admin/toolkit`.

    Check the impact of SR2023 changes to your routing logic and prepare required modifications. Rebuild and redeploy all BAR files that contain message flows that use FTM SWIFT API nodes.
29. If you have both MSIF and MER facility deployed and want to expand BIC addresses using SWIFTRef ReachPlus for FINplus information, load the corresponding reference data using the Reference Data Utility. See Reference Data Utility command for details.
30. If you use Message Entry and Repair (MER) with messages of domain DNIFINPLUS, it is required to perform an additional step to identify outdated messages and templates with the MER message administration utility using the *migratelist* command. Perform a migration action on the detected templates using the MER message administration utility *migtpl* command.

    Note:

    Perform the commands using a special properties file provided for the sole purpose of fix pack migration. Use the `-properties` option when running the Administration Utility with *`inst_dir`*`/run/data/DnqFINPLUS_FP10.xml`.

    ```
    dnpadm migtpl -outputfile finplus.csv -properties /usr/lpp/IBM/ftm/ftmswift/v324/run/data/
    DnqFINPLUS_FP10.xml
    ```

After you finished the fix pack migration, you can continue to convert table spaces that you did not yet migrate to UTS. For details, see “Migrating FTM SWIFT table spaces to universal table spaces (UTS)” on page 37.

Cleaning up

After you have verified that the migrated environment works as expected, and if you are sure that no fallback to the previous level of FTM SWIFT is needed, you can remove obsolete resources:

1. Remove the backed up WebSphere Application Server profiles.
2. Remove the backup of your customized administrative scripts created in step “5” on page 10 (separated file systems) or “5” on page 13 (shared file system):

```
rm -rf ~/admin_scripts_backup
```

3. If your migration starting point was FTM SWIFT fix pack level 3.2.4.7 or 3.2.4.8: Remove the backed up certificate keystore.

Falling back to the previous fix pack level

1. Stop all sessions and services you use.
2. Stop all FTM SWIFT application servers.
3. Stop all FTM SWIFT related message flows.
4. Stop all FTM SWIFT message brokers.
5. Recover the customization system.
6. Roll back the IBM Installation Manager update of the fix pack.
7. Share your files from the installation system with the customization and runtime system, if applicable.
8. If your migration starting point was FTM SWIFT fix pack level 3.2.4.7 or fix pack level 3.2.4.8: Restore the backup of your certificate keystore, for example:

```
cp -p /your_backup_directory/ftmswift_keystore.jks /var/ftmswift_v324/run/ftmswift_keystore.jks
```

9. Restore configuration and security data as described in Restoring configuration and security data.
10. If your migration starting point was FTM SWIFT fix pack level 3.2.4.7 or fix pack level 3.2.4.8: Revert the FTM SWIFT database objects to their previous state.
    a. Open file *deployment_dir*/*instance*/admin/dnicommon_inst_sp_fb02.ddl.
    b. Adapt *USERNAME* and *PASSWORD* in the CONNECT statement.
    c. Replace DNIvOLDINSTPATH with your FTM SWIFT installation directory.
    d. Save your changes.
    e. Use the Db2 command line processor to execute the statements in the file:

```
java com.ibm.db2.clp.db2 +c -td# -z fp10_fallback_jar.log -svf dnicommon_inst_sp_fb02.ddl
```

    Proceed with the file *dnicommon_inst_fb02.ddl*:
    a. Open file *deployment_dir*/*instance*/admin/dnicommon_inst_fb02.ddl.
    b. Adapt *USERNAME* and *PASSWORD* in the CONNECT statement.
    c. Save your changes.
    d. Use the Db2 command line processor to execute the statements in the file:

```
java com.ibm.db2.clp.db2 +c -td# -z fp10_fallback_jar.log -svf dnicommon_inst_fb02.ddl
```

    If you have service bundle DNFRMR deployed:
    a. Open file *deployment_dir*/*instance*/admin/dnfrmr_inst_fb03.ddl.
    b. Adapt *USERNAME* and *PASSWORD* in the CONNECT statement.
    c. Save your changes.
    d. Use the Db2 command line processor to execute the statements in the file:

```
java com.ibm.db2.clp.db2 +c -td# -z fp10_fallback_rmrfb03.log -svf dnfrmr_inst_fb03.ddl
```

    If you have service bundle DNFEFAS deployed:

a. Open file *deployment_dir*/*instance*/admin/dnfefas_inst_fb02.ddl.
b. Adapt *USERNAME* and *PASSWORD* in the CONNECT statement.
c. Save your changes.
d. Use the Db2 command line processor to execute the statements in the file:

```
java com.ibm.db2.clp.db2 +c -td# -z fp10_fallback_efasfb02.log -svf dnfefas_inst_fb02.ddl
```

11. If you have service bundle DNFEFAS deployed revert the FTM SWIFT database objects to their previous state.
    a. Open file *deployment_dir*/*instance*/admin/dnfefas_inst_fb03.ddl.
    b. Adapt *USERNAME* and *PASSWORD* in the CONNECT statement.
    c. Save your changes.
    d. Use the Db2 command line processor to execute the statements in the file:

```
java com.ibm.db2.clp.db2 +c -td# -z fp10_efasfb03.log -svf dnfefas_inst_fb03.ddl
```

12. If your migration starting point was FTM SWIFT fix pack level 3.2.4.9:
    Revert the FTM SWIFT database objects to their previous state.
    a. Open file *deployment_dir*/*instance*/admin/dnicommon_inst_sp_fb03.ddl.
    b. Adapt *USERNAME* and *PASSWORD* in the CONNECT statement.
    c. Replace DNIvOLDINSTPATH with your FTM SWIFT installation directory.
    d. Save your changes.
    e. Use the Db2 command line processor to execute the statements in the file:

```
java com.ibm.db2.clp.db2 +c -td# -z fp10_fallback_jar.log -svf dnicommon_inst_sp_fb03.ddl
```

13. If your migration starting point was FTM SWIFT fix pack level 3.2.4.8:
    a. Open file *deployment_dir*/*instance*/admin/dnicommon_inst_fb03.ddl.
    b. Adapt *USERNAME* and *PASSWORD* in the CONNECT statement.
    c. Save your changes.
    d. Use the Db2 command line processor to execute the statements in the file:

```
java com.ibm.db2.clp.db2 +c -td# -z fp10_fallback_instfb03.log -svf
dnicommon_inst_fb03.ddl
```

14. If your migration starting point was FTM SWIFT fix pack level 3.2.4.7:
    a. Open file *deployment_dir*/*instance*/admin/dnicommon_inst_fb04.ddl.
    b. Adapt *USERNAME* and *PASSWORD* in the CONNECT statement.
    c. Save your changes.
    d. Use the Db2 command line processor to execute the statements in the file:

```
java com.ibm.db2.clp.db2 +c -td# -z fp10_fallback_instfb04.log -svf
dnicommon_inst_fb04.ddl
```

15. Restart your WLM application environment:

```
V WLM,APPLENV=applenv,REFRESH
```

16. Restart all FTM SWIFT message brokers.
17. Deploy previous FTM SWIFT BAR files:

```
. /var/ftmswift_v324/run/dniprofile
dniczbap -cmd prepare -update old -deploy [-broker broker_name]
```

18. Verify the deployed BAR files:

```
dniczbap -cmd list
```

The deployment was successful if the displayed versions contain the fix pack that was your migration starting point.

19. Re-activate FTM SWIFT accounting if you use the SIPN FIN or FMT FIN service.
20. Restart all FTM SWIFT related message flows.
21. Restore the IBM WebSphere Application Server profile backups.
22. Restart all FTM SWIFT application servers.
23. Restart all sessions and services.
24. Restore the backup of your customized administrative scripts created in step “5” on page 10 (separated file systems) or “5” on page 13 (shared file system):

```
rm -rf /var/ftmswift_v324/cus/depdata/INST1/admin/*
cp -p ~/admin_scripts_backup/* /var/ftmswift_v324/cus/depdata/INST1/admin/
```

25. Restore the instance administration data set *prefix.DNIvINST.ADMIN* members created in step “6” on page 10 (separated file systems) or “6” on page 13 (shared file system).

Re-migrating after a previous fallback

After you fall back to an earlier level, plan for re-migration only after you have identified the reason for the fallback and have corrected the problem.

To re-migrate, follow the steps described in this readme document.

Maintenance tasks

The following sections provide detailed instructions for selected installation steps of a fix pack. Refer to "Installing FTM SWIFT 3.2.4.10 – Update an existing installation" on page 9 to find out which steps you have to perform and to determine the sequence.

Ensure that no customization operation is pending

When you apply maintenance fixes to FTM SWIFT, no customization operation must be pending. That is, all previously prepared deployment instructions were carried out and the CDP **implement** command was used before you can apply an update.

To check that all previous CDD changes were implemented using the CDP:

1. Log on to z/OS UNIX on the customization system as a customizer (UCUST1).
2. Enter the following command:

   ```
   inst_dir/admin/bin/dnicdpst -i instance -cdefs cust_defs_dir
   ```

   where:

   inst_dir
   The FTM SWIFT installation directory

   instance
   The name of the FTM SWIFT instance

   cust_defs_dir
   The name of the customization definitions directory as specified in the CDP ini file, for example: `/var/ftmswift_v324/cus/defs`
3. Check whether the response indicates that a customization operation is still pending.
4. If a operation was pending in customization mode (dnicdp), implement it before continuing.
5. If a operation was pending in migration mode (dnicdpm):
   - Ensure that you have not yet shared the new files contained in this or any other product update with the customization system.
   - Implement the pending operation before continuing.

Note: Ensure that no changes are made to the currently implemented CDD until the migration for the current product update has been completely finished.

Ensure that no configuration or security administration change is pending

When you apply maintenance fixes to FTM SWIFT, no configuration or security administration changes must be pending.

To ensure that all configuration administration changes have been deployed and that all security administration changes have been approved:

1. Log on to z/OS UNIX on the runtime system as a system configuration administrator (SA1).
2. Run the `dniprofile` by entering:

   ```
   . /var/ftmswift_v324/run/dniprofile
   ```
3. Enter the following commands:

```
dnicli -s DNI_SYSADM -ou SYSOU -c "list -ou % -qo amorz"
dnicli -s DNI_SYSADM -ou SYSOU -c "list -cos % -qo amorz"
dnicli -s DNI_SYSADM -ou SYSOU -c "list -ct % -qo amorz"
```

4. Check that each list command did result in 'No [OU/COS/CT] match search criteria'.
5. Log on to z/OS UNIX on the runtime system as a security administrator (UA1).
6. Run the `dniprofile` by entering:

```
. /var/ftmswift_v324/run/dniprofile
```

7. Enter the following commands:

```
dnicli -s DNI_SECADM -ou SYSOU -c "list -ro % -qo mor"
dnicli -s DNI_SECADM -ou SYSOU -c "list -rg % -qo mor"
```

8. Check that each list command did result in 'No roles/role groups found that match specified criteria'.
9. Enter the following command for each OU:

```
dnicli -s DNI_SECADM -ou OU -c "list -user % -qo mor"
```

10. Check that each list command did result in 'No users found that match specified criteria'.

Note: Ensure that no changes are made to configuration and security administration until the migration for the current product update has been completely finished.

Use IBM Installation Manager to install the fix pack

Transfer the ZIP file (that you downloaded from Fix Central) in binary mode to the FTM SWIFT installation system, for example, to directory `/usr/lpp/InstallationManagerRepository/HSWS324`.

After you have successfully installed the fix pack using IBM Installation Manager, perform the tasks described in the following sections:

1. “Permissions of the installed files” on page 23
2. “Sharing files with the customization and runtime systems” on page 24
3. “Setting up the use of shared library regions” on page 24
4. “Granting access permissions to FTM SWIFT users” on page 24

IBM Installation Manager offers different modes. The following two sections provide examples using wizard mode (graphical user interface or web) or command line driven installations. Choose one of the IBM Installation Manager modes.

Install a fix pack using wizard mode

To install a fix pack using wizard mode:

1. Start the IBM Installation Manager in graphical user interface or web mode
2. Add the fix pack repository:
   a. Go to **File** > **Preferences** > **Repository** > **Add repository**
   b. Enter the path of the FTM SWIFT fix pack repository file, for example:

```
/usr/lpp/InstallationManagerRepository/HSWS324/Ftm_Swift_Repo.zip
```

   c. Click **OK**
3. Test the repository connection
4. Close the Preferences dialog
5. In the IBM Installation Manager main window, click **Update**
6. Select the package group of the FTM SWIFT installation to update with the fix pack

7. Click **Next**
8. Ensure the correct fix pack is displayed and selected
9. Click **Next**
10. Accept the license agreement
11. Click **Next**
12. Review the summary information and click **Update**
13. Click **Finish**
14. Close the IBM Installation Manager:
    - In graphical user interface mode, click **File** > **Exit**
    - In web mode, click **File** > **Stop server**

Install a fix pack using command line mode

To install a fix pack on the command line:

1. Go to the Installation Manager tools directory, for example:

```
cd /InstallationManager/bin/eclipse/tools
```

2. Check what is currently installed for FTM SWIFT:

```
./imcl listInstalledPackages -long | grep com.ibm.dni
```

   The output includes a line for the installed fix pack. There may be additional lines for installed iFixes. All lines have the format:

```
inst_dir : package_id : name : version
```

   Note the value for *inst_dir*, which is identical in all lines of the output.

3. Run the following command:

```
./imcl install com.ibm.dni.v324
-installationDirectory inst_dir -repositories fix_pack_repo
-acceptLicense
```

   where

   inst_dir
   is the value determined in step “2” on page 23

   fix_pack_repo
   is the FTM SWIFT repository file `Ftm_Swift_Repo.zip`, for example: `/usr/lpp/InstallationManagerRepository/HSWS324/Ftm_Swift_Repo.zip`.

4. Verify the installation result by issuing the following command:

```
./imcl listInstalledPackages -long | grep com.ibm.dni
```

   The output includes the version of the installed fix pack, for example `3.2.4.1` for fix pack 1. Ensure that this version matches the fix pack you are currently installing.

Permissions of the installed files

After installing FTM SWIFT, the installer (UIM1) becomes the owner of the files in the installation system. The owner group is the default group of the installer, for example, DNIINST.

Table 3 on page 24 shows the access permissions of the installed files.

Table 3. Permissions of the installed files			
Directory	**Owner permissions**	**Owner group permissions**	**Other permissions**
/usr/lpp/IBM/ftm/ftmswift/v324	r w x	r - x	r - x
/usr/lpp/IBM/ftm/ftmswift/v324/admin	r w x	r - x	- - -
/usr/lpp/IBM/ftm/ftmswift/v324/run	r w x	r - x	- - -
/usr/lpp/IBM/ftm/ftmswift/v324/iFix	r w x	r - x	- - -

After you install FTM SWIFT, check the owner and the owning group of the product directory and its contents. You might need to adjust these to correspond to the users and groups you chose for your system. When sharing the data between different systems, make sure that all required users and groups have access to this data.

Sharing files with the customization and runtime systems

After installation, make the HFS or zFS directories, that contain the installed product, available to the customization and runtime systems.

If the customization and runtime systems are different, the best way to share the FTM SWIFT data is to define it as a shared HFS or zFS. How to do this is described in *UNIX System Services: Planning*.

If you use other sharing techniques, for example, Network File System (NFS), you can share the complete product directory.

The deployment directory of the customization system must be accessible from each runtime system.

Setting up the use of shared library regions

To reduce the amount of memory required by the broker execution groups, and to decrease startup times, you can use shared library regions. For more information, refer to the section Customizing UNIX System Services on z/OS in the IBM Documentation for IBM Integration Bus.

- Ensure that you have at least read access to the BPX.FILEATTR.SHARELIB FACILITY class. This is required for you to be able to issue the **extattr** command with the +l option.
- To set up the use of shared address spaces and shared libraries for the FTM SWIFT modules of all brokers, issue each of the following commands once:

```
extattr +l inst_dir/run/bin/*
extattr +l inst_dir/run/lib/*
extattr +l inst_dir/run/lil64/*
```

Granting access permissions to FTM SWIFT users

This description assumes that you use the following group names:

- DNIADMIN
- DNILPP

To ease access for these groups, issue the following commands in a z/OS® UNIX shell:

```
chgrp -R DNIADMIN inst_dir/admin
chgrp -R DNILPP inst_dir/run
chmod 755 inst_dir
chmod -R 750 inst_dir/admin
chmod -R 750 inst_dir/run
chmod -R 755 inst_dir/iFix
```

This gives the users in each of the specified groups access to the specified directories and all their subdirectories.

Table 4. Required access permissions to the customization programs, runtime programs, and data				
Directory	**Owner permissions**	**Owner group permissions**	**Other permissions**	**Owner group**
inst_dir	r w x	r - x	r - x	DNIINST
inst_dir/admin	r w x	r - x	- - -	DNIADMIN
inst_dir/run	r w x	r - x	- - -	DNILPP
inst_dir/iFix	r w x	r - x	r - x	DNIINST

Update customization definition data, and create deployment instructions and vehicles

FTM SWIFT maintenance may require to update resources for an instance. The customization definition program (CDP) detects which resources are affected and prepares the necessary deployment data.

To execute the CDP in migration mode:

1. Log on to z/OS UNIX on the customization system as a customizer (UCUST1).
2. Change to the customization file system, for example:

   ```
   cd /var/ftmswift_v324/cus
   ```

3. Run your customization profile:

   ```
   . ./dnicus_instance
   ```

4. Start the CDP in migration mode and use the following commands to migrate customization data:

   ```
   dnicdpm -i instance
   > export cdd/instance_FPxxxx.cdd
   > import cdd/instance_FPxxxx.cdd
   > prepare
   ```

   where

   instance
   The name of the FTM SWIFT instance.

   xxxx
   The version of the fix pack, for example 3241.

   deployment_dir
   The name of the customization deployment directory, for example: /var/ftmswift_v324/cus/depdata

   This step updates the customized administrative scripts in the directory *deployment_dir*/*instance*/admin. It generates deployment instructions and record it in the file *deployment_dir*/*instance*/*timestamp*/instructions.txt. Dependent on the fix pack migration it generates the deployment data and vehicles.

5. Implement the customization definition data and quit the CDP session:

   ```
   > implement
   ```

   When the message "DNIZ9013I: If you continue, the current CDD will be overwritten by a new CDD." is displayed enter 'y' to continue.

   ```
   > quit
   ```

Prepare BAR files for manual deployment

If you want to use the Toolkit or `mqsideploy` command to manually deploy the updated BAR files, you can customize them as soon as you have shared the FTM SWIFT installation directory's `run/flows` subdirectory with the message broker runtime system.

To customize BAR files for manual deployment:

1. Ensure that the updated BAR files are available.

   If your installation and runtime systems are different:
   Share the `run/flows` subdirectory of the FTM SWIFT installation directory from the installation system with the runtime system.

   If your installation and runtime systems are identical:
   Install the update using IBM® Installation Manager as described in "Use IBM Installation Manager to install the fix pack" on page 22 during the switching phase.
2. On the runtime system where the message broker runs, log on as IBM Integration Bus administrator (UWMBA1).
3. Run the `dniprofile` by entering:

   ```
   . /var/ftmswift_v324/run/dniprofile
   ```
4. Create a sub-directory `ftmswift_`*`xxxx`* where *xxxx* is the version of the fix pack. You need read and write access and it must have at least 50 MB of free space. This is the directory in which dniczbap will store the customized BAR files.
5. Issue the following command to let the BAP identify the BAR files that are to be updated and customize them:

   ```
   dniczbap -cmd prepare -update new -dir output_dir
   ```

   where *output_dir* represents the directory you created in step "4" on page 26.

   Each customized BAR file in the output directory has a name of the form:
   `instance``.`*`broker`*`.`*`exec_group`*`.`*`BAR_file`*`.bar` where

   instance
   The name of your FTM SWIFT instance.

   broker
   The name of the broker to which the BAR file is to be deployed.

   exec_group
   The name of the execution group to which the BAR file is to be deployed.

   BAR_file
   The name of the BAR file as provided by FTM SWIFT.
6. Transfer, in binary mode, the customized BAR files in the output directory to the system where you need to deploy them, for example your Toolkit system.
7. If you use the Toolkit to deploy the customized BAR files, import them now into your workspace.

Stop all FTM SWIFT related message flows

FTM SWIFT related message flows are based on FTM SWIFT provided IBM Integration Bus plugins. To ensure that both are updated before new messages are processed you need to stop the flows.

FTM SWIFT related message flows include:

- Flows provided by FTM SWIFT
- Flows you developed based on FTM SWIFT APIs

You can use either the BAP, the Toolkit or the command `mqsistopmsgflow` to stop message flows provided by FTM SWIFT. For flows that you have developed you have to use the Toolkit or `mqsistopmsgflow`.

To use the BAP to stop the message flows provided by FTM SWIFT on each broker server:

1. Ensure that your brokers and execution groups are still running.
2. On the runtime system, log on to z/OS UNIX as IBM Integration Bus administrator (UWMBA1).
3. Run the `dniprofile` by entering:

```
. /var/ftmswift_v324/run/dniprofile
```

4. Issue the following command to stop all message flows provided by FTM SWIFT on the current broker:

```
dniczbap -cmd stop
```

Verifying the installation of the database routines

To verify the installation of the database routines:

1. On the runtime system, log on to z/OS UNIX as a Db2 administrator (UDB2ADM1).
2. Ensure that you have access to a Java™ runtime environment.
3. Run the dniprofile by entering:

```
. /var/ftmswift_v324/run/dniprofile
```

4. Issue the **dnimaintinfo** command, for example:

```
dnimaintinfo -dsn MYDB -schema DNI
```

   For details about the **dnimaintinfo** command, see Maintenance Information command.
5. Examine the output and ensure that the following message is displayed:

```
DNID0001I Jar file version verification successful
```

   If you did not assign the **DNFFIN** service bundle (SVB) to any OU, the output should be:

```
DNID0015E JAR file 'dnfcdrtn.jar' for jarId 'dnfcdrtn' is either not installed or has an
unexpected version.
```

Deploy BAR files

During the switching phase you need to update the message flows running in IBM Integration Bus. If you use multiple broker servers, you must perform the following steps for each.

If you have created customized BAR files as described in “Prepare BAR files for manual deployment” on page 26, use the Toolkit or `mqsideploy` now to deploy them.

To use the BAP to automatically customize and deploy updated BAR files:

1. Ensure that your brokers and execution groups are running.
2. On the runtime system, log on to z/OS UNIX as IBM Integration Bus administrator (UWMBA1).
3. Run the `dniprofile` by entering:

```
. /var/ftmswift_v324/run/dniprofile
```

4. Ensure that you have at least 50 MB of free space in the current directory.
5. Issue the following command:

```
dniczbap -cmd prepare -update new -deploy -broker brokername
```

The parameter -broker is only required if you use multiple broker servers.

The BAP will identify all BAR files for which the message flows deployed in the broker need to be updated and automatically customize and deploy them.

Re-activate FTM SWIFT accounting

If you use the SIPN FIN or FMT FIN service, re-activate FTM SWIFT accounting.

1. Issue the following z/OS console commands:

```
F broker,CHANGEFLOWSTATS A=YES,E='eg',f='DNF_ILS_ACK',C=ACTIVE,B=BASIC,O='XML'
F broker,CHANGEFLOWSTATS A=YES,E='eg',f='DNF_ILS_FIN',C=ACTIVE,B=BASIC,O='XML'
```

where:

broker
The name of the broker.

eg
The name of the execution group.

If you deployed the above mentioned bar files to multiple execution groups, repeat the steps for each execution group in which the bar files are deployed.

Restart all FTM SWIFT related message flows

After the updated message flows have been deployed as described in “Deploy BAR files” on page 27 you need to restart your message flows.

You can use either the BAP, the Toolkit or the command `mqsistartmsgflow` to start message flows provided by FTM SWIFT. For flows that you have developed you have to use the Toolkit or `mqsistartmsgflow`.

To use the BAP to start the message flows provided by FTM SWIFT on each broker server:

1. Ensure that your brokers and execution groups are running.
2. On the runtime system, log on to z/OS UNIX as IBM Integration Bus administrator (UWMBA1).
3. Run the `dniprofile` by entering:

```
. /var/ftmswift_v324/run/dniprofile
```

4. Issue the following command to start all message flows provided by FTM SWIFT on the current broker:

```
dniczbap -cmd start
```

Recover the customization system

Recover former service bundles, and restore the current definition directory and the deployment directory for administrative resources *deployment_dir*/*instance*/admin.

1. Log on to z/OS UNIX on the customization system as a customizer (UCUST1).
2. Change to the customization file system, for example:

```
cd /var/ftmswift_v324/cus
```

3. Run your customization profile:

```
. ./dnicus_instance
```

4. Start the CDP in migration mode and use the following commands to recover customization data:

```
dnicdpm -i instance
> recover
```

where *instance* is the name of the FTM SWIFT instance.

Roll back the IBM Installation Manager update of the fix pack

Use the roll back feature of IBM Installation Manager to remove an update and revert to a previous fix pack of FTM SWIFT.

After having reverted to a previous version of FTM SWIFT, follow the instructions in:

1. “Permissions of the installed files” on page 23
2. “Sharing files with the customization and runtime systems” on page 24
3. “Setting up the use of shared library regions” on page 24
4. “Granting access permissions to FTM SWIFT users” on page 24

IBM Installation Manager offers different modes. The following two sections are examples using wizard mode (graphical user interface or web) or command line driven roll backs. Choose one of the IBM Installation Manager modes.

Roll back using wizard mode

To roll back a fix pack using wizard mode:

1. Start Installation Manager in graphical user interface or web mode.
2. Click **Roll Back**.
3. Select the package group of FTM SWIFT and click **Next**.
4. Select the fix pack level to roll back to.
5. Click **Next**.
6. Review the summary information and click **Roll Back**.
7. Click **Finish**.
8. Close the Installation Manager:
   - In graphical user interface mode, click **File** > **Exit**.
   - In web mode, click **File** > **Stop server**

Roll back using command line mode

To roll back FTM SWIFT to the previously installed fix pack on the command line:

1. Go to the Installation Manager tools directory, for example:

   ```
   cd /InstallationManager/bin/eclipse/tools
   ```

2. Run the following command:

   ```
   ./imcl rollback com.ibm.dni.v324
   ```

3. Verify the roll back result:

   ```
   ./imcl listInstalledPackages -long | grep com.ibm.dni
   ```

   The output includes the version of the installed fix pack, for example `3.2.4.1` for fix pack 1. Ensure that this version matches the fix pack you are rolling back to.

Update an SAG Add-On

If a fix pack contains an update of SAG Add-On, use IBM Installation Manager to install the update. How to obtain the Installation Manager repository is described in Installing the SAG Add-On / Pre-installation steps. You do not need to stop the SAG in order to update the SAG Add-On.

To update an SAG Add-On:

1. Stop the SAG Add-On.

   How to do this depends on the operating system of your SAG workstation:

   - On AIX®: Stop the SAG Add-On subsystems as described in Stopping an SAG Add-On on AIX
   - On RHEL x86: Stop the SAG Add-On service as described in Stopping an SAG Add-On on RHEL x86
   - On Windows: Stop the SAG Add-On service as described in Starting, stopping, or displaying the status of an SAG Add-On

   If the SAG Add-On cannot be stopped, stop the SAG Add-On process manually. How to do this depends on the operating system of your SAG workstation and is described here:

   - For AIX: Killing the SAG Add-On process on AIX (use only if the process is deadlocked)
   - For RHEL x86: Killing the SAG Add-On process on RHEL x86 (use only if the process is deadlocked)
   - For Windows: Starting, stopping, or displaying the status of an SAG Add-On
2. Create a backup of your SAG Add-On profile `dnfcssao.cfg` that is located in the SAG Add-On runtime directory:

   - On AIX and RHEL x86: `/var/ftmswift_v324/sao`
   - On Windows: `%PROGRAMDATA%\ftmswift_v324\sao`

   Note: Do not store the backup file in the SAG Add-On runtime directory, but in a different location.
3. Uninstall the currently installed version of the SAG Add-On using IBM Installation Manager.
4. Install the SAG Add-On with the new fix pack level using IBM Installation Manager.
5. Copy the backup of the SAG Add-On profile `dnfcssao.cfg` (that you created in step “2” on page 30) to the SAG Add-On runtime directory:

   - On AIX and RHEL x86: `/var/ftmswift_v324/sao`
   - On Windows: `%PROGRAMDATA%\ftmswift_v324\sao`
6. Start the SAG Add-On.

Prepare the migration of configuration entities

FTM SWIFT maintenance may require to update configuration entities. The program dnfczmlc compares your current configuration with the target configuration. If it detects differences it creates CLI command files which will contain the configuration migration statements to bring your environment to the target configuration.

To prepare the migration of configuration entities:

1. If your installation and runtime systems are different:

   Share the `run/data` subdirectory of the FTM SWIFT installation directory from the installation system with the runtime system.
2. On the runtime system, log on to z/OS UNIX as the system configuration administrator (SA1), and run the profile for your runtime environment by entering:

   ```
   . /var/ftmswift_v324/run/dniprofile
   ```
3. Create a sub-directory `ftmswift_`*xxxx* where *xxxx* is the version of the fix pack, for example 3241.
4. Switch to this directory and enter the following command:

```
dnfczmlc -i instance [-dual YES|NO] [-to timeout]
```

where

-i *instance*
The name of the FTM SWIFT instance.

-dual YES|NO
Specifies whether files are to be created for a system that uses dual authorization for SYSOU. The default is -dual YES. Specify -dual NO only if dual authorization is turned off for both DNI_SYSADM and DNI_SECADM in SYSOU at the time when the created files are executed. Whether dual authorization is switched on or off for other OUs is irrelevant.

-to timeout
The number of milliseconds that the CLI waits for a response to this command before it issues an error message. The default is 100000 (100 seconds). It must be a whole number between 20000 and 9999999.

The command dnfczmlc lists the CLI command files that it created in the current directory, for example:

```
Generating the command files for migration ...
The following files are generated and need to be executed for migration:

Seq  User    File
---  ------  ------------------------------
001  Any UA  dnfczmlc_1_ua_rem_ro_all.cli
002  Any SA  dnfczmlc_2_sa_ent_all.cli
003  Any UA  dnfczmlc_3_ua_cre_ro_all.cli

DnfInfo: Script /opt/IBM/ftm/swift/v324/run/bin/dnfczmlc completed successfully.
```

Note: The command dnfczmlc starts a long-running task that might take several minutes to complete.

5. Save the output of dnfczmlc which tells you the sequence and the user ID you have to use later when you submit the CLI command files in “Migrate the configuration entities” on page 31.

Migrate the configuration entities

FTM SWIFT maintenance may require to update configuration entities. In section “Prepare the migration of configuration entities” on page 30 you created the required CLI command files that now need to be executed.

To migrate the configuration entities:

1. For each CLI command file listed in the output of dnfczmlc in “Prepare the migration of configuration entities” on page 30, log on to z/OS UNIX as the user specified for the current file.

   The user IDs are:

   1st, 2nd, or Any SA
   The first system configuration administrator (SA1), the second system configuration administrator (SA2), or any system configuration administrator.

   1st, 2nd, or Any UA
   The first user administrator (UA1), the second user administrator (UA2), or any user administrator.

2. Run the profile for your runtime environment by entering:

   ```
   . /var/ftmswift_v324/run/dniprofile
   ```

3. Switch to the sub-directory ftmswift_*xxxx* you created in section “Prepare the migration of configuration entities” on page 30, step “3” on page 30.
4. Run the current CLI command file by issuing the following command:

   ```
   dnicli -s svc -ou SYSOU -cft file | tee -a dnfczmlc_cli_xxxx.log
   ```

   where:

svc

DNI_SYSADM
For files executed by a system configuration administrator.

DNI_SECADM
For files executed by a security administrator.

file
The CLI command file name, for example dnfczmlc_5_sa_cre_ct_com.cli.

xxxx
The version of the fix pack, for example 3241.

5. Check the log file to see if any error occurred.

Providing Db2 administration modules

For some tasks, you need to issue SQL statements or DSN or Utility commands that are contained in a specific administration module. To copy these modules from the HFS deployment directory to MVS™ data sets on the runtime system:

1. Log on to TSO as a customizer (UCUST1).
2. Check whether the following partitioned target data sets are already on the runtime system and, if not, allocate them:
   - The instance administration data set, which is the data set in which instance-level administration modules that operate on objects that are shared by different OUs of the same FTM SWIFT instance are stored. Give the data set a name of the form:

     ```
     prefix.DNIvINST.ADMIN
     ```

   - The OU administration data set, which is the data set in which OU-specific administration modules that operate within the scope of a single OU are stored. You must define one such data set per OU. Give each of the data sets a name of the form:

     ```
     prefix.DNIvINST.ADMIN.ou
     ```

   where:

   prefix
   The data set prefix, for example, FTMDEP.FTMSW324.

   ou
   The name of the OU addressed by the modules in the data set.

   Use the following DCB values:

   ```
   DSORG = PO
   RECFM = FB
   LRECL = 80
   ```

3. Open a z/OS UNIX session.
4. Change to the following directory:

   ```
   deployment_directory/DNIvINST/admin
   ```

   This directory contains the deployment data for the ADMIN resource class of the instance server and the following scripts, which copy the administration modules from the HFS file system to the MVS data sets that were allocated in step “2” on page 32:

   dniczcpa
   Copies instance-level administration modules of FTM SWIFT to your instance administration data set.

dnfczcpa
Copies instance-level administration modules of the SIPN FIN service to your instance administration data set. This module is needed (and is available) only if the SIPN FIN service bundle was assigned to one or more server-OU combinations.

<ou>_dniczcpo
Copies OU-specific administration modules of FTM SWIFT to your OU administration data set.

5. Edit each of these scripts:
   - In the copy commands, replace the prefix FTMDEP.FTMSW324 with the prefix specified in step "2" on page 32.
   - Search for placeholders that were not assigned values during customization (these will begin with the characters DNFv or DNIv), and replace each of them with an appropriate value.
6. Run the scripts from the deployment directory. For example, if the name of your instance is INST1:
   a) Change to the directory:

```
/var/ftmswift_v324/cus/depdata/INST1/admin
```

   b) To run **dniczcpa** enter:

```
./dniczcpa
```

   This script makes the following modules available in your instance administration data set (FTMDEP.FTMSW324.*DNIvINST*.ADMIN):

   DNIMZCFO
   Invokes REORG utility for selected table spaces of configuration and security administration tables.

   DNIMZCFR
   Restores, to the FTM SWIFT configuration and security administration tables, the consistent set of image copies previously generated by module DNIMZCFS.

   DNIMZCFS
   Creates a consistent set of image copies that contain the data in the FTM SWIFT configuration and security administration tables.

   DNIMZORO
   Invokes REORG utility for selected table spaces of service bundle DNFEFAS.

   DNIMZOST
   Starts RUNSTATS for table spaces of service bundle DNFEFAS.

   DNIMZREO
   JCL sample procedure to remove the PENDING state on a table.

   DNIMZRST
   Starts RUNSTATS for the FTM SWIFT configuration and security administration tables.

   c) To run **dnfczcpa** enter:

```
./dnfczcpa
```

   This script makes the following modules available in your instance administration data set (FTMDEP.FTMSW324.*DNIvINST*.ADMIN):

   DNFCZFUS
   Manipulates the catalog statistics of tables DNF_OAMS and DNF_IAMS.

   DNFMZBND
   Binds the Db2 application package of the finite state machine (FSM) of the SWIFTNet FIN Daemon (SFD).

   DNFMZRBD
   Rebinds the Db2 application package of the FSM of the SWIFTNet FIN Daemon.

DNFMZREO
Reorganizes the table spaces of tables DNF_OAMS and DNF_IAMS, including the associated control tables.

DNFMZRST
Starts RUNSTATS for table spaces of service bundle DNFFIN.

DNFMZRS1
Starts RUNSTATS for table spaces of service bundle DNFFIN. The statistics information in the catalog is not updated.

d) To run each script with a name of the form *ou*_dniczcpo enter:

```
./ou_dniczcpo
```

Each of these scripts makes the following modules available in the corresponding OU administration data set (FTMDEP.FTMSW324.*DNIvINST*.ADMIN.*ou*):

DNICDCAP
Creates the Db2 objects of the partitioned message audit log for the default partitioning scheme.

DNICDATR
Creates the trigger to activate the data integrity framework for the partitioned message audit log.

DNICDVAP
Creates a view on the partitioned message audit log to allow applications to access it as if it were a non-partitioned message audit log.

Saving configuration and security data

To generate the consistent set of image copies:

1. On the system where the Db2 subsystem resides, log on to TSO as a Db2 administrator (UDB2ADM1).
2. In the data set *prefix*.*instance*.ADMIN, locate and edit the member DNIMZCFS as follows:
   a) Adapt the job card to your needs.
   b) Locate and replace all occurrences of the placeholder DNIxICPF with the prefix of the image copy data sets (up to 8 characters).
3. Submit the job.
4. Check the SYSPRINT output. Ensure that the highest return code was 0. If the job was customized as suggested, after successful completion, the image copies are stored in data sets with names of the following form:

   ```
   prefix.db2_subsystem_id.DNI.database.tablespace.suffix
   ```

   where *prefix*, *db2_subsystem_id*, *database*, *tablespace*, and *suffix* represent values set in DNIMZCFS.

Restoring configuration and security data

To restore configuration and security data:

1. On the system where the Db2 subsystem resides, log on to TSO as a Db2 administrator (UDB2ADM1).
2. In the data set with the name *prefix*.*instance*.ADMIN, locate and edit the member DNIMZCFR as follows:
   a) Adapt the job card to your needs.
   b) Locate and replace all occurrences of the placeholder DNIxICPF, including those in the header, with the prefix of the image copy data sets (up to 8 characters).
   c) The header of DNIMZCFR contains SQL statements similar to those shown below.

```
SELECT HEX(START_RBA)
     FROM  SYSIBM.SYSCOPY
     WHERE     DBNAME = (SELECT DBNAME
                         FROM SYSIBM.SYSTABLES
                         WHERE     NAME    = 'DNI_CT'
                               AND CREATOR = 'schema'
                        )
           AND TSNAME = (SELECT TSNAME
                         FROM SYSIBM.SYSTABLES
                         WHERE     NAME    = 'DNI_CT'
                               AND CREATOR = 'schema'
                        )
           AND DSNAME LIKE 'prefix.db2_subsystem_id.DNI.dbname.%'
   ;
```

The schema name, prefix, Db2 subsystem ID, and database name are already set to those of the image copy data set.

Execute these statements to retrieve the RBA or LRSN recorded in SYSIBM.SYSCOPY for the consistent set of image copies created in “Saving configuration and security data” on page 34.

d) To specify the TOLOGPOINT in the RECOVERY statement, replace the dashes by the RBA or LRSN retrieved in step “2.c” on page 34.

3. Submit the job.
4. Check the SYSPRINT output. Ensure that the highest return code was 4. This warning return code is caused by the indexes being in REBUILD PENDING status after the RECOVER statements, but before the REBUILD statements, were processed. After successful completion the configuration and security tables are reset to their original and previously saved state.

Update the IBM Integration Toolkit workstation

To install the new versions of the Toolkit resources, follow the instructions listed in:

- Transferring the FTM SWIFT Toolkit resources
- Installing the FTM SWIFT Eclipse plug-ins

If you use FTM SWIFT sample message flows as foundation for your own flow development follow the instructions provided in Using the sample routing flows.

Otherwise, continue with the instructions provided in:

- Importing FTM SWIFT sample projects
- Importing the message sets and sample routing flows

Additionally, if you use FTM SWIFT message set projects containing XML schema definitions for your own flow development follow the instructions provided in: Importing XSD files for SWIFT message payloads.

Furthermore, you have to rebuild and redeploy your message flows if they are based on the FTM SWIFT API.

Migrating FTM SWIFT table spaces to universal table spaces (UTS)

Note: You can skip this chapter, if the starting point of the migration to FTM SWIFT 3.2.4 fix pack 10 was a new FTM SWIFT instance created with a minimum fix pack 2. All related table spaces were created as UTS table spaces when this instance was created, and no migration is required.

This section describes how you migrate the table spaces of an existing FTM SWIFT instance to UTS.

After you have finished the installation and migration to FTM SWIFT 3.2.4 fix pack 10, you can migrate the segmented table spaces of existing instances to partition-by-growth universal table spaces (UTS) except of the following table spaces:

- DNFASP (used by tables for ASP data)
- DNFOMI1S and DNFOMI2S (used by some MSIF tables)
- DNFOMWM (used by message warehouse tables)
- DNFRMDP (used by tables for RMA authorisations)
- DNFTS04 (used by table DNF_MSGS for SIPN FIN and FMT FIN processing)
- DNQEMSG (used by MER tables)
- The table spaces identified by the following placeholders:
  - DNFvMWF, DNIvMWH, DNFvMWM, DNFvMWOF, and DNFvMWX (used by message warehouse tables)
  - DNIvAMB, DNIvAMH, and DNIvAUM (used by message audit tables)
  - DNIvAMCO and DNIvAMP (used by tables for partitioned message audit log)
  - DNIvAUU (used by user audit tables)
  - DNIxTSHI (used by tables in the history database)

You can migrate the table spaces at your own pace; for example, you can migrate all table spaces at once, or one table space after the other. To check which table spaces are not yet migrated to UTS, you can use the **-verify_ts** action of the maintenance information command dnimaintinfo, for example:

```
dnimaintinfo -verify_ts -dsn DNIDB -schema DNI -uid UDB2ADM1 -pw password
```

To migrate the FTM SWIFT table spaces, a FTM SWIFT downtime is required. During the migration phase all FTM SWIFT services, sessions, application servers, and message brokers must be stopped.

To migrate the table spaces to UTS, you can do either of the following:

Using vehicles provided by FTM SWIFT

In this case, proceed as described in “Performing migration of table spaces using FTM SWIFT vehicles” on page 38.

If a problem occurs after a table space was migrated, you can fall back as described in “Falling back” on page 46. To be able to fall back to the original table space, you must backup the appropriate table data as described in “Backing up” on page 46 before you start to migrate a table space.

Using Db2 functional level 508

You can use the new Db2 functional level 508 to migrate the FTM SWIFT table spaces listed in column **Table spaces** of Table 6 on page 41.

- For more information about using the new functional level 508, see https://www.ibm.com/docs/en/db2-for-zos/12?topic=d1fl-function-level-508-activation-enabled-by-apar-ph29392-october-2020#db2z_fl_v12r1m508__e101.
- For information about the feature *Moving tables from multi-table table spaces to partition-by-growth table spaces (UTS)*, see https://www.ibm.com/docs/en/db2-for-zos/12?topic=ats-moving-tables-from-multi-table-table-spaces-partition-by-growth-table-spaces.

Note: If you decide to migrate the table spaces by using Db2 functional level 508, you must use exactly the same table space names as listed in column **Table spaces** of Table 6 on page 41. For more information, contact your IBM service representative.

Performing migration of table spaces using FTM SWIFT vehicles

To migrate the table spaces to UTS by using the vehicles provided by FTM SWIFT, proceed as follows:

1. Preparation:
   a. On the runtime system, log on as a Db2 administrator (UDB2ADM1).
   b. Set up the usage of the Db2 command line processor (CLP) for the migration modules:
      i) Create a properties file `clp.properties` for the Db2 command line processor that contains the following line defining the FTM SWIFT connection alias:

      ```
      FTMDBALIAS=DNIvDBHOST:DNIvDBPORT/DNIvDSN,USER,PASSWORD
      ```

      where:

      DNIvDBHOST
      The hostname

      DNIvDBPORT
      The database port

      DNIvDSN
      The name of the Db2 location containing the runtime database

      USER
      The user ID of the database administrator (UDB2ADM1)

      PASSWORD
      The password of the database administrator

      Make sure that only you, the Db2 administrator, has read access to this properties file.

      ii) Execute the following command, and ensure to execute it every time you use the CLP for migration purposes (for example, by adding it to your profile):

      ```
      export CLPPROPERTIESFILE=DNIvCLPPATH/clp.properties
      ```

      where *DNIvCLPPATH* represents the path to the properties file that you created in step "1.b.i" on page 38.
   c. Prepare the stored procedures used by the UTS table space migration:
      i) Set up the WLM and authorizations in order to execute the stored procedure DSNUTILU provided by Db2.

      Note: If you are using RACF authorization, you must set up a RACF group and assign the Db2 administrator (UDB2ADM1) to that group.

      For more information about the setup of the DSNUTILU, see https://www.ibm.com/docs/en/db2-for-zos/12?topic=db2-dsnutilu.

      ii) Create the stored procedures DNI_CALL_REORG and DNI_CHECK_TSTYPE:
         a) Prepare DNI_CALL_REORG to use the Db2 REORG utility:

         The default z/OS data set pattern used for sequential copy dataset is

         ```
         &US..&SSID..UNL.&DB..&TS..D&JD.&MI.
         ```

         where:

         - *&US* is the invoking user id
         - *&SSID* is the subsystem id

- *UNL* is a fix qualifier
- *&DB* is the database name
- *&TS* is the table space name
- *D* is the fix prefix of the last qualifier
- *&JD* is the day in year
- *&MI* is the minute in year

If you do not want to use this default data set pattern, or if you do not want to use the utility ID DNIUTSREORG, edit the file *deployment_dir*/*DNIvINST*/admin/dnimuts_cre_checkts_sp.ddl and change the data set pattern or utility ID as required.

b) Execute the following command to create the stored procedures DNI_CALL_REORG and DNI_CHECK_TSTYPE:

```
java com.ibm.db2.clp.db2 +c -td# -z dniuts_sp.log -svf deployment_dir/DNIvINST/admin/dnimuts_cre_checkts_sp.ddl
```

2. Migration:

For each service bundle that you have deployed, check Table 6 on page 41 to determine which table spaces you want to migrate to UTS. For each of these table spaces perform the following steps:

a. Backup the table data for the case of a fallback. There are two ways to do this (which are described in “Backing up” on page 46).

If you decided to use the sample JCL **DNIUBAK**:

i) Edit **DNIUBAK**.

ii) Replace the data set prefix placeholder *DNIxICPF* with a value that meets your requirements.

iii) Replace the placeholder *TSLSTDEF* with the file that is specified in column **List definition TSLSTDEF for DNIUBAK** of Table 6 on page 41 for the table space to be migrated.

iv) Run the job and verify the output for success.

v) For recover purposes, obtain the image copy dataset name of the table space from the job output. Search for message DSNU1038I and note the value of its parameter **DSN** (shown in the example job output below, where **DNFOFD** is the table space name).

```
DSNU1038I   310 14:34:03.04 DSNUGDYN - DATASET ALLOCATED.  TEMPLATE=SYSCOPY
DSN=UDB2ADM.DC11.COPYF.DSN1.DNFOFD.D31034
```

b. In Table 6 on page 41, identify the DDL file that is specified for the table space to be migrated. This DDL file resides in directory *deployment_dir*/*DNIvINST*/admin. Run this DDL file by using the Db2 CLP.

For example, issue the following CLP command to migrate the table space DNIOU:

```
java com.ibm.db2.clp.db2 +c -t -z dnimuts_ou.log -svf deployment_dir/DNIvINST/admin/dnimuts_ou.ddl
```

If the command was executed successfully, the following message is displayed:

```
DNIB1001I : UTS table space migration successful for tablespace
```

where *tablespace* is the table space that was migrated.

If a Db2 error occurred, check the SQL statement that caused the problem. Resolve the problem and rerun the migration DDL. If the problem relates to DNI_CALL_REORG or DNI_CHECK_TSTYPE, inspect the following table for more information on how to proceed:

Table 5. SQLSTATE values issued by the stored procedures DNI_CALL_REORG and DNI_CHECK_TSTYPE

SQLSTATE	Explanation	Required action
99TS0	Stored procedure DNI_CALL_REORG was issued with inconsistent input parameters.	Check the invocation of DNI_CALL_REORG within the migration module and adapt the input parameters. Run the procedure DNI_CALL_REORG again.
99TS1	The specified table space was not found.	Check the invocation of the stored procedure DNI_CHECK_TSTYPE and verify if the correct table space name and database name are provided. Run the procedure DNI_CHECK_TSTYPE again.
99TS2	The specified table space does not have the expected type and was not successfully migrated to a universal table space.	Check the REORG output and fix the errors. Then, run the procedure DNI_CALL_REORG again.

c. Execute the DIC command build if you have activated the data integrity framework, and if the output of step "2.b" on page 39 contains the following message:

```
DNIB1002I : If the data integrity framework is active,
            you have to run DIC Build for the following table:
            table
```

where *table* is the table that must be specified for the **-table** parameter of the DIC command build.

For example, if you did run the DDL file `dnimuts_ou.ddl` to migrate table space DNIOU used by database table DNI_OU, execute the following command:

```
dnpdic -build -Djava.security.policy=/var/ftmswift_v324/run/ftmswift.policy
       -passphrase @/var/ftmswift_v324/run/passphrase.stash
       -dsn DSN1 -schema DNI
       -table DNI_OU
```

The following table provides, per service bundle, the following information:

- The DDL files to be used for migration
- The table spaces that are migrated by a DDL file
- Whether invocation of DIC command build is required after migration
- The DDL files to be used for fallback (if required)
- The list definition file to be used for data backup in step "2.a.iii" on page 39

Table 6. UTS DDL files for service bundles

Service bundle	DDL file required for migration	Table spaces	DIC build required?	DDL file required for fallback	List definition TSLSTDEF for DNIUBAK
DNICOMMON	dnimuts_common.ddl	DNICOS DNICT DNISCOM DNICTA DNICOSRE DNITIMER DNIDBSTA DNIDBHIS DNICNTRL DNISESS DNICUR DNICTY DNIBICI DNIRDU DNIACC DNIUPR DNIRDMBI DNIRDMCT DNIRDMCU		dnimuts_common_fb.ddl	uts_ld_common
	dnimuts_cos_ct_con_rel.ddl	DNICOSCC DNICOSCA DNICOSCB	✓	dnimuts_cos_ct_con_rel_fb.ddl	uts_ld_cos_ct_con_rel
	dnimuts_ct_attr_value.ddl	DNICTAV DNICTAVA DNICTAVB	✓	dnimuts_ct_attr_value_fb.ddl	uts_ld_ct_attr_value
	dnimuts_event.ddl	DNIEVENT DNIEVENA DNIEVENB	✓	dnimuts_event_fb.ddl	uts_ld_event
	dnimuts_ou.ddl	DNIOU DNIOUA DNIOUB	✓	dnimuts_ou_fb.ddl	uts_ld_ou

Table 6. UTS DDL files for service bundles (continued)

Service bundle	DDL file required for migration	Table spaces	DIC build required?	DDL file required for fallback	List definition TSLSTDEF for DNIUBAK
DNICOMMON *(continued)*	dnimuts_rg_role_rel.ddl	DNIRGR DNIRGRA DNIRGRB	✓	dnimuts_rg_role_rel_fb.ddl	uts_ld_rg_role_rel
	dnimuts_ro_ct_attr_rel.ddl	DNIROA DNIROAA DNIROAB	✓	dnimuts_ro_ct_attr_rel_fb.ddl	uts_ld_ro_ct_attr_rel
	dnimuts_role_resolved.ddl	DNIROR DNIRORA DNIRORB	✓	dnimuts_role_resolved_fb.ddl	uts_ld_role_resolved
	dnimuts_role.ddl	DNIROLE DNIROLEA DNIROLEB	✓	dnimuts_role_fb.ddl	uts_ld_role
	dnimuts_rolegroup.ddl	DNIRG DNIRGA DNIRGB	✓	dnimuts_rolegroup_fb.ddl	uts_ld_rolegroup
	dnimuts_user_resolved.ddl	DNIURV DNIURVA DNIURVB	✓	dnimuts_user_resolved_fb.ddl	uts_ld_user_resolved
	dnimuts_user.ddl	DNIUSR DNIUSRA DNIUSRB	✓	dnimuts_user_fb.ddl	uts_ld_user
	dnimuts_usr_rg_rel.ddl	DNIURG DNIURGA DNIURGB	✓	dnimuts_usr_rg_rel_fb.ddl	uts_ld_usr_rg_rel
	dnimuts_usr_role_rel.ddl	DNIUSRRO DNIUSRRA DNIUSRRB	✓	dnimuts_usr_role_rel_fb.ddl	uts_ld_usr_role_rel

Table 6. UTS DDL files for service bundles (continued)

Service bundle	DDL file required for migration	Table spaces	DIC build required?	DDL file required for fallback	List definition TSLSTDEF for DNIUBAK
DNFEFAS	dnfmefasuts.ddl	DNFOMI3S DNFOFE DNFOLH DNFOTO DNFORE1 DNFOFI DNFOFO DNFOIW		dnfmefasuts_fb.ddl	uts_ld_efas
	dnfmefasuts_dnfo_config_data.ddl	DNFOCD1 DNFOCD1A DNFOCD1B	✓	dnfmefasuts_dnfo_config_data_fb.ddl	uts_ld_config_data
	dnfmefasuts_dnfo_fsm_download.ddl	DNFOFD DNFOFDA DNFOFDB	✓	dnfmefasuts_dnfo_fsm_download_fb.ddl	uts_ld_fsm_download
	dnfmefasuts_dnfo_fsm_rcv_msg.ddl	DNFORM1 DNFORM1A DNFORM1B	✓	dnfmefasuts_dnfo_fsm_rcv_msg_fb.ddl	uts_ld_fsm_rcv_msg
	dnfmefasuts_dnfo_fsm_receive.ddl	DNFOFR DNFOFRA DNFOFRB	✓	dnfmefasuts_dnfo_fsm_receive_fb.ddl	uts_ld_fsm_receive
	dnfmefasuts_dnfo_fsm_send.ddl	DNFOFS DNFOFSA DNFOFSB	✓	dnfmefasuts_dnfo_fsm_send_fb.ddl	uts_ld_fsm_send
	dnfmefasuts_dnfo_fsm_snd_msg.ddl	DNFOSM DNFOSMA DNFOSMB	✓	dnfmefasuts_dnfo_fsm_snd_msg_fb.ddl	uts_ld_fsm_snd_msg
	dnfmefasuts_dnfo_fsm_state.ddl	DNFOFA DNFOFAA DNFOFAB	✓	dnfmefasuts_dnfo_fsm_state_fb.ddl	uts_ld_fsm_state

Table 6. UTS DDL files for service bundles (continued)

Service bundle	DDL file required for migration	Table spaces	DIC build required?	DDL file required for fallback	List definition TSLSTDEF for DNIUBAK
DNFEFAS *(continued)*	dnfmefasuts_dnfo_lob_data.ddl	DNFOLD DNFOLDA DNFOLDB	✓	dnfmefasuts_dnfo_lob_data_fb.ddl	uts_ld_lob_data
	dnfmefasuts_dnfo_msg_part.ddl	DNFOMP DNFOMPA DNFOMPB	✓	dnfmefasuts_dnfo_msg_part_fb.ddl	uts_ld_msg_part
	dnfmefasuts_dnfo_mwh_data.ddl	DNFOMD DNFOMDA DNFOMDB	✓	dnfmefasuts_dnfo_mwh_data_fb.ddl	uts_ld_mwh_data
DNFFIN	dnfmfinuts.ddl	DNFFLTSS		dnfmfinuts_fb.ddl	uts_ld_fin
	dnfmfinuts_iams.ddl	DNFIAMS DNFIAMSA DNFIAMSB	✓	dnfmfinuts_iams_fb.ddl	uts_ld_iams
	dnfmfinuts_oams.ddl	DNFTS01 DNFTS01A DNFTS01B	✓	dnfmfinuts_oams_fb.ddl	uts_ld_oams
DNFFINCI	<OU>_dnfmzuts.ddl (OU specific file)	DNFvFCA (OU specific placeholder)		<OU>_dnfmzuts_fb.ddl (OU specific file)	<OU>_uts_ld_finci (OU specific file)
DNFFMTFIN	dnfmfmtfinuts.ddl	DNFPFR		dnfmfmtfinuts_fb.ddl	uts_ld_fmtfin
DNFVERIF	dnfmsigverifuts.ddl	DNFVRQ DNFVCY DNFVTM DNFVMSG		dnfmsigverifuts_fb.ddl	uts_ld_verif

Table 6. UTS DDL files for service bundles (continued)

Service bundle	DDL file required for migration	Table spaces	DIC build required?	DDL file required for fallback	List definition TSLSTDEF for DNIUBAK
DNFRMA	dnfrmauts.ddl	DNFRMY DNFRMSS		dnfrmauts_fb.ddl	uts_ld_rma
	dnfrmauts_rmah.ddl	DNFRMAH DNFRMAHA DNFRMAHB	✓	dnfrmauts_rmah_fb.ddl	uts_ld_rmah
	dnfrmauts_rmqh.ddl	DNFRMQH DNFRMQHA DNFRMQHB	✓	dnfrmauts_rmqh_fb.ddl	uts_ld_rmqh
	dnfrmauts_rmqs.ddl	DNFRMQS DNFRMQSA DNFRMQSB	✓	dnfrmauts_rmqs_fb.ddl	uts_ld_rmqs
DNFRMR	dnfrmruts.ddl	DNFRMRL DNFRMAL DNFRMTS		dnfrmruts_fb.ddl	uts_ld_rmr
DNPAO	dnpmzuts.ddl	DNPAOLS		dnpmzuts_fb.ddl	uts_ld_aols
DNQER	dnqmzuts.ddl	DNQEMD		dnqmzuts_fb.ddl	uts_ld_er
DNQPRINT	dnqmputs.ddl	DNQPQUE DNQPORD DNQPMSG DNQPCNT		dnqmputs_fb.ddl	uts_ld_print

Backing up

If you migrate a table space by using FTM SWIFT vehicles, you must backup the appropriate table data before. You can do this in one of the following ways:

Using private or local procedure
Backup the table data, the table, and the table space definitions on your own.

Using the FTM SWIFT sample
Use the sample JCL **DNIUBAK** provided in the data set FTMDEP.FTMSW324.*DNIvINST*.ADMIN.

Note: For this backup method it is necessary that the definitions of the original table spaces, data tables, and indexes were not changed in comparison to FTM SWIFT provided definitions.

To use the sample JCL **DNIUBAK**, copy the appropriate file from the HFS deployment directory into an z/OS data set:

1. On the customization system, log on to TSO as a customizer (UCUST1).
2. Edit the script *deployment_directory*/*DNIvINST*/admin/dniczcpu according to its usage description.
3. Execute the script:

```
./dniczcpu
```

Falling back

Falling back from UTS means to return to Db2 segmented table spaces. This might be necessary if, after migration, you encounter severe problems with a migrated table space that can best be resolved by reverting it to its original. You have only to revert those table spaces for which you encounter severe problems, and you can keep the migrated UTS table spaces that are working.

To fall back a migrated UTS table space to a segmented table space:

- If you backed up your database by your own, restore this backup.
- If you used the sample JCL **DNIUBAK** as described in “Backing up” on page 46, do the following to restore the original table space:
  1. On the runtime system, log on as a Db2 administrator (UDB2ADM1).
  2. Run the corresponding fallback DDL module that is specified in Table 6 on page 41 and that resides in directory *deployment_dir*/*DNIvINST*/admin.

     For example, issue the following CLP command to fall back from the UTS table spaces DNIOU, DNIOUA, and DNIOUB:

```
java com.ibm.db2.clp.db2 +c -t -z dnimuts_ou_fb.log -svf deployment_dir/DNIvINST/admin/dnimuts_ou_fb.ddl
```

     If the command was executed successfully, a message like the following one is displayed:

```
DNIB1011I : Fallback module successful for tablespaces
```

     where *tablespaces* is the list of table spaces that were processed.
  3. Recover the data of the table space:
     a. Edit the sample JCL **DNIUREC** in FTMDEP.FTMSW324.*DNIvINST*.ADMIN.
     b. Replace all occurrences of placeholder *DNIxTS* with the table space that you want to revert.
     c. Replace placeholder *IMGCOPYDS* with the data set name (**DSN**) that you noted during migration in step “2.a.v” on page 39.

        Note: To find out which table space was migrated by which DDL file, inspect Table 6 on page 41.
     d. Run the job.

Cleaning up

As soon as all table spaces listed in column **Table spaces** of Table 6 on page 41 are migrated, the Db2 administrator can do the following:

- Delete the backup images of the table spaces that were created when using sample JCL **DNIUBAK** as described in step “2.a” on page 39.
- Delete the file `clp.properties` that was created in step “1.b.i” on page 38.

However, do not drop the stored procedures DNI_CALL_REORG and DNI_CHECK_TSTYPE that were created in step “1.c.ii.2” on page 39.

Copyright and trademark information

http://www.ibm.com/legal/copytrade.shtml

Document change history

Date	Description of change
2023-08-31	Initial publication date

Product Number: 5655-FTB