MERVA ESA Components

MERVA USE Administration Guide

Version 4 Release 1

SH12-6338-03

MERVA ESA Components

MERVA USE Administration Guide

Version 4 Release 1

SH12-6338-03

Note!

Before using this information and the product it supports, be sure to read the general information under "Appendix E. Notices" on page 103.

Fourth Edition, October 2001

This edition applies to Version 4 Release 1 of IBM MERVA ESA Components (5648-B30) and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces SH12-6338-02.

Changes to this edition are marked with a vertical bar.

Acknowledgments

Information relating to SWIFT services is provided by permission of the Society for Worldwide Interbank Financial Telecommunication s.c., La Hulpe, Belgium.

Contents

About This Book

This book describes how to administer the SWIFT User Security Enhancement (USE) functions of the following IBM licensed programs:

- Message Entry and Routing with Interfaces to Various Applications USE & Branch for Windows NT
- Message Entry and Routing with Interfaces to Various Applications USE & Branch for Windows NT with SWIFT Link

This book also contains information that is required by administrators of MERVA ESA.

The main purposes of the USE functions are to:

- Gain access to SWIFT services by means of integrated circuit cards (ICCs) and card readers, instead of using paper-based LOGIN and SELECT tables.
- Enable the user to generate bilateral keys within secure card readers (SCRs) and exchange the keys with correspondents over the SWIFT network, instead of by written correspondence.

Note: The panels and examples may differ from those actually displayed by your system, because:

- You may not be authorized to use certain functions. Options that you are not authorized to use are grayed out.
- This book describes the system as delivered. However, most of the functions are customizable using the Customization program.

Who Should Read This Book

This book is written for persons who are responsible for administering USE functions. It is assumed that the reader is familiar with:

- The MERVA operation and administration functions, in particular, maintenance of the user file. Refer to the *MERVA USE & Branch for Windows NT User's Guide*.
- The MERVA installation and customization functions as described in the *MERVA USE & Branch for Windows NT Installation and Customization Guide*.

  Note: Ensure that the USE related customization steps, especially the definition of USE message headers, have been performed before using the functions described in this book.
- SWIFT terminology as defined in the *S.W.I.F.T. User Handbook* and *S.W.I.F.T. Use Planning Guide* published by the Society for Worldwide Interbank Financial Telecommunication s.c., La Hulpe, Belgium (SWIFT).
- The operation of secure card readers. Refer to the *S.W.I.F.T. Card Readers User Guide*.

Conventions and Terminology Used in This Book

In this book, the following naming conventions apply:

- MERVA USE & Branch for Windows NT is used when the description applies only to MERVA USE & Branch for Windows NT.

- MERVA USE & Branch for Windows NT with SWIFT Link is used when the description applies only to MERVA USE & Branch for Windows NT with SWIFT Link.
- MERVA is used when the description applies to MERVA USE & Branch for Windows NT and MERVA USE & Branch for Windows NT with SWIFT Link at the same time.
- MERVA ESA is used when the description applies only to MERVA ESA.

Chapter 1. Introduction

This chapter describes the way that administrative tasks are grouped, and provides background information about card readers.

The Administrative Tasks

The administrative tasks for USE are divided into three groups:

- **User security officer (USOF) tasks**, for example administering integrated circuit cards (ICCs) and card readers. To carry out these tasks, select the **SWIFT USE USOF** program group from the MERVA menu. These tasks are described in more detail in “Chapter 2. Tasks of the User Security Officer (USOF)” on page 3.
- **User key management officer (UKMO) tasks**, for example preparing for and maintaining the bilateral key exchange (BKE) service. To carry out these tasks, select the **SWIFT USE UKMO** program group from the MERVA menu. These tasks are described in more detail in “Chapter 3. Tasks of the User Key Management Officer (UKMO)” on page 29.
- **User tasks**, for example pregenerating session keys to implement the Secure Login and Select (SLS) service and perform LOGIN and SELECT requests. To carry out these tasks, select the **Communication** program group for the SWIFT SLS Administration program. These tasks are described in more detail in “Chapter 4. Tasks of the User” on page 89.

Using Card Readers with USE Functions

A number of USE functions require the presence of a USE card reader. Two different types of card readers exist:

BCR Basic Card Reader. This type of card reader is not tamper resistant, and supports the SLS service only.

SCR Secure Card Reader. This type of card reader contains sophisticated tamper resistant mechanisms to protect the secret information held on it, and supports both the SLS and BKE services.

Certain functions can only be carried out on an SCR.

Table 1 lists USE functions and shows which type of card reader is required to carry out the functions:

Table 1. USE Functions Requiring a Card Reader

Function	SCR	BCR
Cards - Unblock - Set PIN parameters - Set LT access rights	 x x x	
Card sets - Personalize - Update whitelist flag	 x x	
Card Reader Maintenance - Carry out interface test - Read information from card reader	 x x	 x x

Table 1. USE Functions Requiring a Card Reader (continued)

Function	**SCR**	**BCR**
Generate Public Key (RSA)	x	
Secure Transmission Key (STK) - Generate Secure Transmission Key - Install Secure Transmission Key - Activate Secure Transmission Key	 x x x	
Certificate Handling - Create certificates - Delete certificate in card reader	 x x	
SLS administration	x	x
Note: Card readers are also required for some functions of the USE background process (see the *MERVA USE & Branch for Windows NT Installation and Customization Guide*).		

Note: The required card readers can be connected to the local workstation or to a remote workstation. Remote card readers are configured by using a TCP/IP network link.

For more information on how to install and customize TCP/IP for remote USE card readers, refer to the *MERVA USE & Branch for Windows NT Installation and Customization Guide*.

Chapter 2. Tasks of the User Security Officer (USOF)

As a user security officer (USOF) it is your responsibility to:

- Ensure that users can log in to the SWIFT network
- Ensure that every user, including USOFs and UKMOs, has a valid card
- Ensure that card readers are working properly and maintained regularly
- Maintain the security of access to the SWIFT network by updating card parameters and replacing cards, both routinely and in emergencies
- Deal with card reader emergencies

This involves carrying out the following tasks:

- **Administering card readers**

  Card readers read the cards that enable users to gain access to SWIFT services. The USOF:

  - Records details of the card reader in the database after it has been delivered by SWIFT
  - Tests the interface between SWIFT and the card reader
  - Blacklisting card readers whose security might have been compromised
  - Deleting card readers
- **Administering card sets**

  Cards are grouped into card sets. The USOF:

  - Orders card sets from SWIFT and acknowledges their receipt
  - Deletes unneeded card sets or card sets whose security has been compromised
- **Administering cards**

  In addition to the tasks relating to the card set as a whole, the USOF performs certain functions on individual cards. The USOF:

  - Personalizes new cards (unblocks the card, sets the PIN, sets LT access rights
  - Issues each card to a particular user, specifying such aspects as access rights to Logical Terminals (LTs)
  - Records and updates the details of each card in the database
  - Updates the kernel version of cards that require it
  - Invalidates lost or faulty cards
  - Unblocks cards that have been blocked following a number of unsuccessful attempts at entering the personal identification number (PIN) or PINs.

You can use the full support of MERVA for these tasks when the card reader is used in connected mode. MERVA directly updates the cards and also generate all related SWIFT messages.

If you are using card readers in unconnected mode, you can still read the information displayed on the card reader’s display and enter it into the appropriate window, from where the information is stored in the database. Thereafter, if changes occur, you can display the information currently stored in the database and change it as necessary.

Figure 1 shows the program group for the USOF administration tasks.

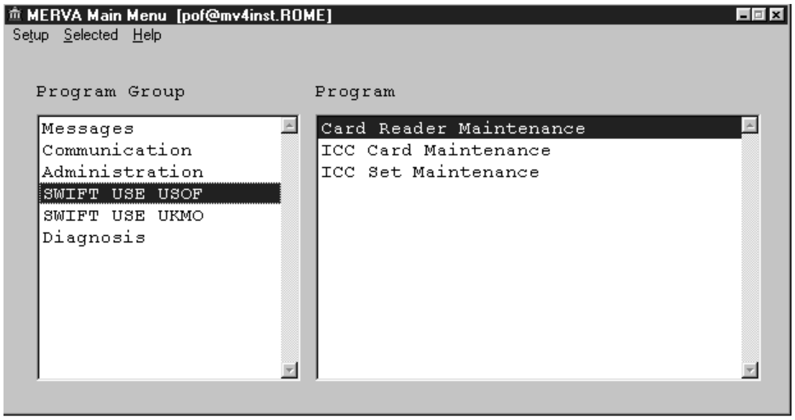


Figure 1. The User Security Officer Program Group

When you are preparing to start using cards and card readers for the first time, work through the programs in the order shown in the following table, beginning with the **Card Reader Maintenance** program.

Program Name	**Administration Function**
Card Reader Maintenance	Administering card readers, see “Administering Card Readers”.
ICC Set Maintenance	Administering card sets, see “Administering Card Sets” on page 8.
ICC Card Maintenance	Administering cards, see “Administering Individual Cards” on page 17.

Administering Card Readers

As USOF, you are responsible for ordering, installing, testing, configuring, and, if necessary, blacklisting card readers.

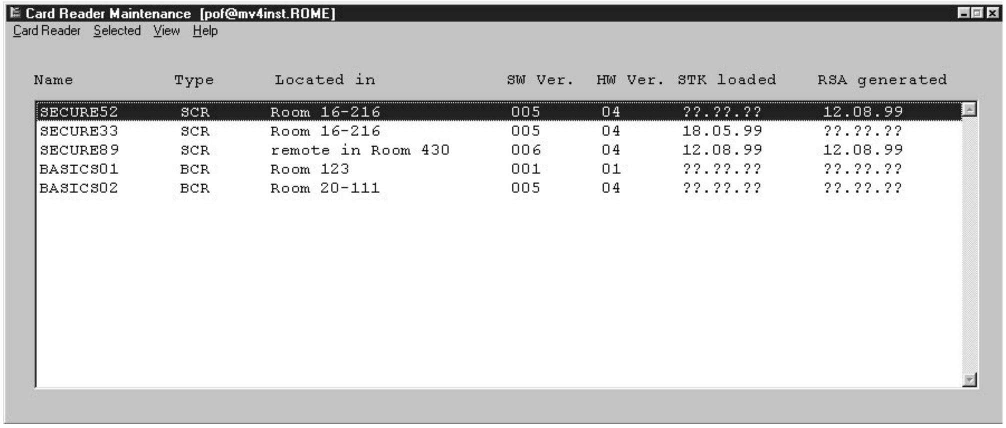
When you receive a card reader from SWIFT and have installed it, you should test that the card reader is working correctly. You can do this using the hardware and software test procedures provided with the card reader and described in the *S.W.I.F.T. Card Readers User Guide*.

Configuring a Card Reader

Your next task is to configure the card reader for use in your MERVA environment. You do this by reading configuration details from the card reader and providing additional information yourself. You enter these details on the Card Reader - Details window, from where they are stored in the database and can be accessed by other USE programs as necessary. You must have registered at least one card reader before you can use the ICC Set Maintenance and ICC Card Maintenance programs.

To record the details of a card reader:

1. Invoke the **Card Reader Maintenance** program from the **SWIFT USE USOF** program group. The list of card readers already known to the system appears, as shown in Figure 2.



Card Reader Maintenance [pof@mv4inst.ROME]
Card Reader Selected View Help

```
Name          Type      Located in               SW Ver.  HW Ver.  STK loaded    RSA generated

SECURE52      SCR       Room 16-216              005      04       ??.??.??      12.08.99
SECURE33      SCR       Room 16-216              005      04       18.05.99      ??.??.??
SECURE89      SCR       remote in Room 430       006      04       12.08.99      12.08.99
BASICS01      BCR       Room 123                 001      01       ??.??.??      ??.??.??
BASICS02      BCR       Room 20-111              005      04       ??.??.??      ??.??.??
```

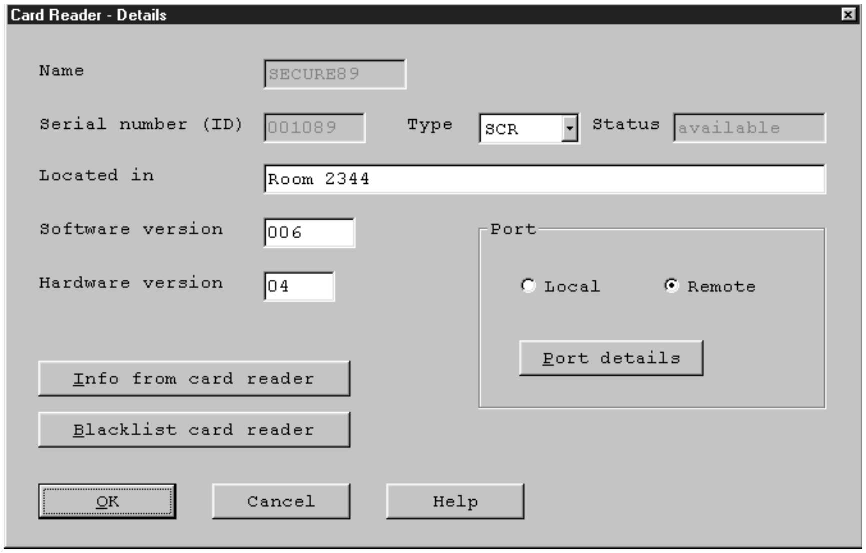
Figure 2. The Card Readers Window

The **STK Loaded** and **RSA Loaded** columns indicate whether the STK and RSA keys are stored in the card reader:

- If the STK or RSA key has been generated on this workstation, the column contains the date that the key was generated.
- If the STK or RSA key has been generated, but on a different workstation or in unconnected mode, **??.??.??** is displayed in the respective column.

  This is because the date that the key was generated is not known in this situation.
- If the STK or RSA key has not yet been generated, the respective column is empty.

2. Select **New card reader** from the **Card reader** pull-down menu. The Card Reader - Details window appears, as shown in Figure 3.



Name: SECURE89

Serial number (ID): 001089 Type: SCR Status: available

Located in: Room 2344

Software version: 006

Hardware version: 04

Port: ○ Local ● Remote

Figure 3. The Card Reader - Details Window

3. In the **Name** field, enter a unique name for the card reader. This field is mandatory.

   You can use up to 8 characters for the name. The first six characters must be letters (A to Z). The last two characters can be either letters or numbers. Embedded blanks are not allowed.
4. In the **Located in** field, enter a comment describing where the card reader is physically located, for example, **Room 2344**.
5. Select whether the card reader is attached directly to the local workstation or via a remote TCP/IP connection.
6. Before you can read details from the card reader, you must check that the communications parameters set on the workstation match those of the card reader itself. To do this, click on **Port details** to display the Port Details window.

   Check that the line speed, parity, and number of data and stop bits used match those of the card reader, which are specified in the *S.W.I.F.T. Card Readers User Guide*.

   Select the **COM** port for a locally attached card reader, for example, **COM1**, or the host name and TCP socked port number for a remotely attached card reader, for example, **usofhost** and **7119**.

   See the *MERVA USE & Branch for Windows NT Installation and Customization Guide* for details on installing and customizing TCP/IP for remote USE card readers.

   Then click on **OK** to return to the Card Reader - Details window.
7. Your workstation can now read the remaining details from the card reader. Click on the **Info from card reader** push button to upload the details from the card reader to the database. The information appears in the Card Reader - Details window.

   If you receive a timeout message, either:
   - The port details are incorrect.
   - The card reader is not attached to the port you have specified.

   You may need to check the card reader cabling and power supply, or use the diagnosis and reference diskette supplied with your workstation to correct the port definition.

   You can also enter the details when the card reader is in unconnected mode. Configure the card reader, as described in the *S.W.I.F.T. Card Readers User Guide* and enter the following values from the card reader’s display into the Card Reader - Details window:
   - The serial number of the card
   - The type of card reader (BCR or SCR)
   - The port details.

   Note: The serial number and card reader type are also printed on the card reader itself.
8. Click on the **OK** push button to save the details and return to the list of card readers.

   The new card reader now appears in the list of card readers.

Testing the Interface

After you have set the required communications parameters for the card reader, you can carry out an interface test to check the physical connection between SWIFT and the card reader. To carry out an interface test:

1. Select **Interface test** from the **Card reader** pull-down menu.
2. Select the **Name** of the card reader to test the interface to. The serial number (**ID**) of the card reader is displayed.
3. Enter a sequence of up to 50 alphanumeric characters of **Test data** to be sent to the card reader.
4. Click on the **Test** push button.

The test data is sent from the workstation to the card reader using the current parameter settings. The card reader transmits the received data back to your workstation. In this way, MERVA can determine whether the interface between the workstation and the card reader is functioning properly.

If the test is not successful, check that:

- The cable from the card reader is attached to the correct port on the workstation
- The port details are correct.

Blacklisting a Secure Card Reader

If you have any reason to believe that the security of an SCR has been compromised, that is, it has been tampered with or mislaid, then the security of the certificates and other secret information stored in the card reader are at risk and you must blacklist the SCR. You do this by requesting SWIFT to add the card reader to a list of unusable card readers. This list is updated regularly by SWIFT and new certificate requests for blacklisted card readers rejected.

Once you have blacklisted an SCR, any BKE messages sent using the certificates stored in the card reader are no longer valid. You must also notify the UKMO to revoke all certificates associated with the public key of that SCR. See “Revoking CVs” on page 43.

To blacklist an SCR:

1. Invoke the **Card Reader Maintenance** program from the **SWIFT USE USOF** program group. The list of card readers known to your system appears.
2. Select the entry for the SCR you want to blacklist.

   Note: No secret information is stored in a basic card reader (BCR), so a tampered with or mislaid BCR does not represent a security risk and does not need to be blacklisted.
3. Select **Blacklist** from the **Selected** pull-down menu. Alternatively, display the Card Reader - Details window (see Figure 3 on page 5) and click on the **Blacklist** push button.
4. Select the **Emitting destination**.

   The emitting destination is necessary to identify the SWIFT destination that sends the message requesting the blacklisting.

   Note: The destinations available for selection are defined on the USE Message Headers window of the MERVA Customization program. See the *MERVA USE & Branch for Windows NT Installation and Customization Guide* for details.

5. Click on the **Blacklist** push button to blacklist the card reader.
6. You are asked to confirm that you want to send an MT090 (Blacklist SCR Request) message to SWIFT Click on the **Yes** push button to create the message and queue it for sending to SWIFT If you select **No**, you must send the message yourself.

The status of the card reader shown on the Card Readers window changes to BLACKLISTED.

If you blacklist a card reader by mistake, you can restore it by deleting the card reader and creating it as a new card reader (select **New Card Reader** from the **Card Reader** pull-down menu).

Deleting a Card Reader

When a card reader is no longer in use, its details can be deleted from the database.

To delete a card reader:

1. Invoke the **Card Reader Maintenance** program from the **SWIFT USE USOF** program group. The list of card readers known to your system appears.
2. Select the entry for the card reader you want to delete.
3. Select **Delete** from the **Selected** pull-down menu.
4. You are asked to confirm the deletion of the card reader. Click on the **Yes** push button to delete the card reader.

Administering Card Sets

As USOF, you are responsible for ordering, issuing, activating, and maintaining UKMO, USOF, and USER cards. Each card belongs to a card set; they can never be used in isolation. Card sets both provide security control for missing cards and ease maintenance, because some administrative tasks can be performed for an entire set instead of individually for each card.

Before starting to manage card sets, you must have configured at least one card reader (see “Configuring a Card Reader” on page 4).

The first step in the management of card sets is to decide how many card sets you require and the composition of the card sets:

- You must order a separate card set for each destination or group of destinations that your financial institution represents.
- You should also order one or more spare sets for each destination. If, for example, a USOF card is lost, SWIFT procedures specify that you replace the entire card set. (You can, however, use a second USOF card in a set to unblock a USOF card.)
- For each set, you need a USOF card for each user security officer, an UKMO card for each user key management officer, and a USER card for each user that is to log in to the SWIFT network.
- It is advisable to order additional cards of each type in case, for example, a card is lost. For example, order three UKMO cards even if your installation has only appointed two user key management officers.
- The smallest card set you can order consists of one USOF card and one USER card.

- An UKMO card is required if your destination acts as a Key Management Authority (KMA) and exchanges its own bilateral keys.
- The maximum number of cards in a set is 99.

The Life Cycle of a Card Set

From the time you order a card set from SWIFT until it is no longer required, each card set passes through a number of statuses. The tasks you can perform on a card set depend on its current status. For example, you can only make a card set available for use (status AVAILABLE) if it currently has the status IN STOCK.

The current status of a card set is shown in the **Status** column of the ICC Sets window.

You can change the status yourself using the View/Change Details window (see “Viewing and Changing Details of a Card Set” on page 16). The possible status values are:

Status **Meaning**

ORDERED The set has been ordered but not yet delivered.

IN STOCK The set has been delivered but cannot be activated because the USOF PIN has not yet arrived.

For security reasons, the PIN for USOF cards is dispatched separately by SWIFT

AVAILABLE The USOF PIN has arrived and the set can be activated for use at any time.

To activate a card set, you send an MT090 (Activate ICC Set Request) message to SWIFT

PENDING ACTIVE

The set is activated for use, but the activation date and time has not been reached. When this date and time is reached, the set becomes current.

CURRENT The set is in use.

PREVIOUS The set has been taken out of use, but is not yet obsolete.

OBSOLETE The set is obsolete. It can no longer be reactivated, and can be physically destroyed and deleted from the database. See “Deleting a Card Set” on page 17 for details of how to do this.

Figure 4 on page 10 illustrates the life cycle of a card set.

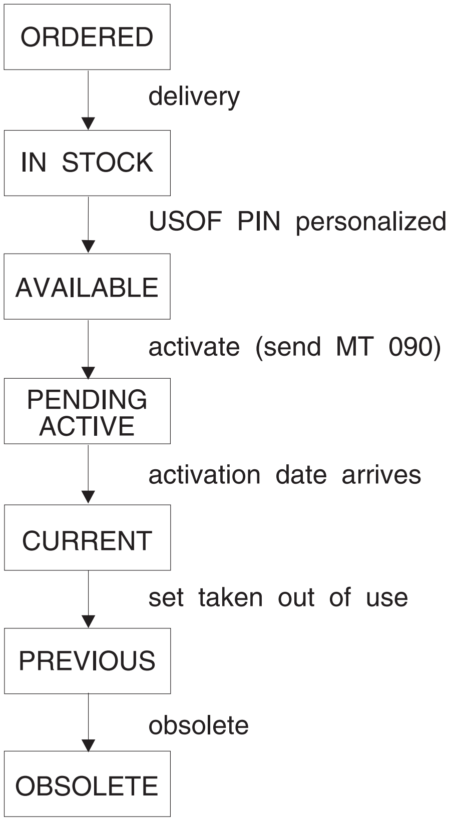


Figure 4. The Life Cycle of a Card Set

Ordering a New Card Set

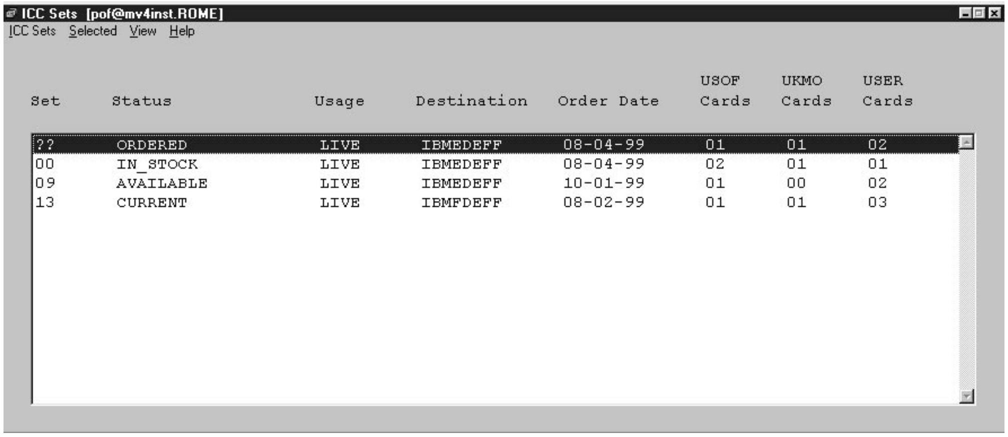
You order card sets using the form supplied by S.W.I.F.T. Order new card sets when:

- You first start working with SWIFT's User Security Enhancements.
- You receive a warning from the SWIFT Security Management Centre (SMC) that the parameters of a set can only be renewed once more.
- You do not have enough available sets. See "Acknowledging the Receipt of a Card Set" on page 12 to find out what "available" means.

Note: To order individual cards to supplement an existing set, see "Ordering Additional Cards for an Existing Set" on page 23.

To register the information about a card set you have ordered:

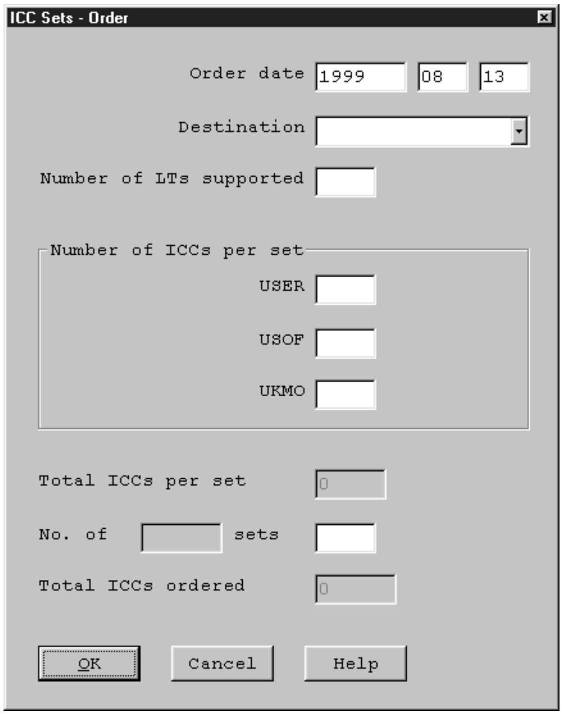
1. Invoke the **ICC Set Maintenance** program from the **SWIFT USE USOF** program group.
2. If more than one card reader has been customized as belonging to this workstation, the Card Reader Selection window is displayed. Select the card reader to use and click on **OK**.
3. The list of sets known to your system appears on the ICC Sets window shown in Figure 5 on page 11.



Set	Status	Usage	Destination	Order Date	USOF Cards	UKMO Cards	USER Cards
??	ORDERED	LIVE	IBMEDEFF	08-04-99	01	01	02
00	IN_STOCK	LIVE	IBMEDEFF	08-04-99	02	01	01
09	AVAILABLE	LIVE	IBMEDEFF	10-01-99	01	00	02
13	CURRENT	LIVE	IBMFDEFF	08-02-99	01	01	03

Figure 5. The ICC Sets Window

4. Select **Order new set/s** from the **ICC Sets** pull-down menu. The ICC Sets - Order window appears. For an example, see Figure 6.



Order date 1999 08 13

Destination

Number of LTs supported

Number of ICCs per set

USER

USOF

UKMO

Total ICCs per set 0

No. of sets

Total ICCs ordered 0

Figure 6. The ICC Sets - Order Window

5. Select the date of ordering the card set. The default is today’s date.
6. Select the SWIFT **Destination** of the card set from the customized list of destinations. Notice that the final digit of the destination determines the type of card set ordered:
   - 0 indicates a test and training set. **TEST** appears in the **Number of sets** field.
   - Any other digit indicates a live set. **LIVE** appears in the **Number of sets** field.
7. Specify the number of LTs supported by the selected destination.

8. Fill in the number of USER, USOF, and UKMO cards that are to be included in the set.
9. Specify the number of live or test and training sets ordered.

The values for total number of cards per set and total number of cards ordered are calculated and entered in the fields automatically.

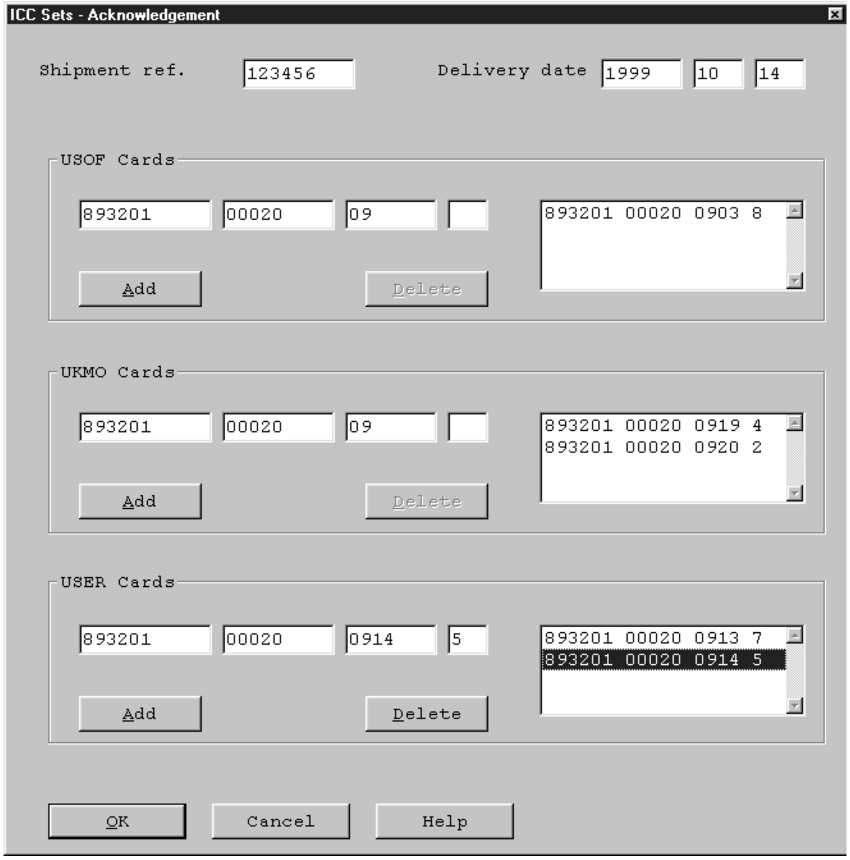
10. Click on the **OK** push button to leave the ICC Sets - Order window and return to the list of ICC sets.

The status of the card set is now ORDERED. A set number has not yet been allocated, so this set appears with two question marks (??) in the **Set** column of the ICC Sets window.

Acknowledging the Receipt of a Card Set

When a card set has been delivered to you, you must acknowledge its receipt before the PINs are sent to you. To do this:

1. Invoke the **ICC Set Maintenance** program from the **SWIFT USE USOF** program group. The list of sets known to your system appears.
2. Select the card set with status ORDERED from the list on the ICC Sets window.
3. Select **Acknowledge receipt** from the **Selected** pull-down menu. The ICC Sets - Acknowledgement window appears, as shown in Figure 7.



ICC Sets - Acknowledgement

Shipment ref. 123456 Delivery date 1999 10 14

USOF Cards

893201 00020 09

893201 00020 0903 8

UKMO Cards

893201 00020 09

893201 00020 0919 4
893201 00020 0920 2

USER Cards

893201 00020 0914 5

893201 00020 0913 7
893201 00020 0914 5

Figure 7. The ICC Sets - Acknowledgement Window

4. Enter the card shipment reference number of the set, which appears on the SWIFT form delivered with the card set.
5. Enter the date that the set was delivered by SWIFT
6. Enter the number printed on the first USOF card. The first 13 characters of the number are the same for each card in a set. Therefore, as you type, these characters are copied to the corresponding fields for the UKMO and USER cards to make entering those numbers easier. However, you must enter the final three digits of the number printed on the card yourself.
7. Click on the **Add** push button, or press the space bar, to add the card number to the list in the column on the right.
8. Repeat the previous two steps for all USOF cards in the set.
9. Repeat the same procedure for all UKMO and USER cards in the set.
10. When you have entered the card numbers of all the cards in the set, click on the **OK** push button.
11. You are asked to confirm that you want to create an MT085, Delivery Confirmation, message.Click on the **Yes** push button to create the message. An MT085 message is created and queued for sending to SWIFT

    If this is the first time you have ordered cards from SWIFT and you cannot log in to the SWIFT network to send the message, select **No** and send the message yourself (by post, for example).

The status of the card set shown on the ICC Sets window changes to IN STOCK.

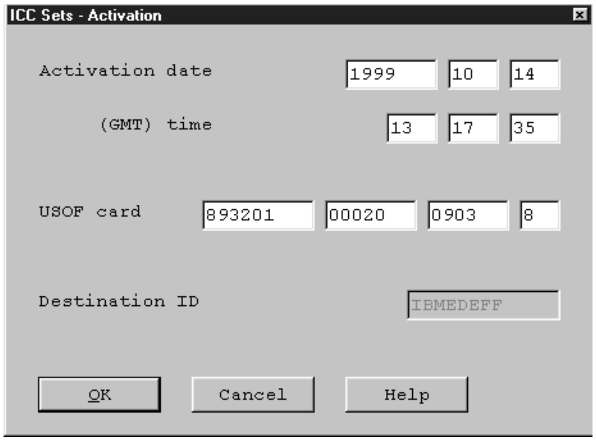
Activating a Card Set

Before any of the cards delivered to you can be used, you must create a current set by activating one of the available sets, that is, a set that has never been in use before. You activate an available set:

- When you first start to use the SLS and BKE services.
- If the set parameters (the kernel version or the whitelist flag) for the current set need to be changed and cannot be further updated.
- If a USER or USOF card from the current set has been mislaid and the set parameters can no longer be updated.

To activate an available set:

1. Invoke the **ICC Set Maintenance** program from the **SWIFT USE USOF** program group. The list of sets known to your system appears.
2. Select the card set to activate. The status of the card set must be AVAILABLE.
3. Select **Activate** from the **Selected** pull-down menu. The ICC Sets - Activation window appears, as shown in Figure 8 on page 14.



Activation date 1999 10 14

(GMT) time 13 17 35

USOF card 893201 00020 0903 8

Destination ID IBMEDEFF

Figure 8. The ICC Sets - Activation Window

4. The **USOF card** number and the **Destination ID** are displayed. Specify the date and time from which to activate the set.
5. Click on the **OK** push button.
6. You are asked to confirm that you wish to create an MT090, Activation Request message, and queue it for sending to SWIFT Click on the **Yes** push button to create the message. An MT090 is created and sent to SWIFT requesting activation of the set at the date and time you specified.

   If you click on **No**, you must send a message to SWIFT yourself (by post, for example) proposing a date and time for activating the card set. This is the case when activating the very first set in an installation. (At this stage, you do not have a USER card to be able to log in and send the MT090.)

The status of the set is changed to PENDING ACTIVE. When the activation time is reached, the set becomes CURRENT.

Only one card set can be current for a destination at any one time. When the activation date and time arrive, the set that was in use is no longer valid and its status changes to PREVIOUS. From the user’s point of view, the card set is replaced with a new card set, and the previous one can no longer be used.

Personalizing New Cards

Before a card can be used, it must be prepared, or *personalized*. Different types of cards have different preparation steps:

- For USOF cards, you must set the PIN parameters.
- For UKMO cards, you must set the PIN parameters and unblock the card.
- For USER cards, you must set the PIN parameters and logical terminal (LT) access rights, and unblock the card.

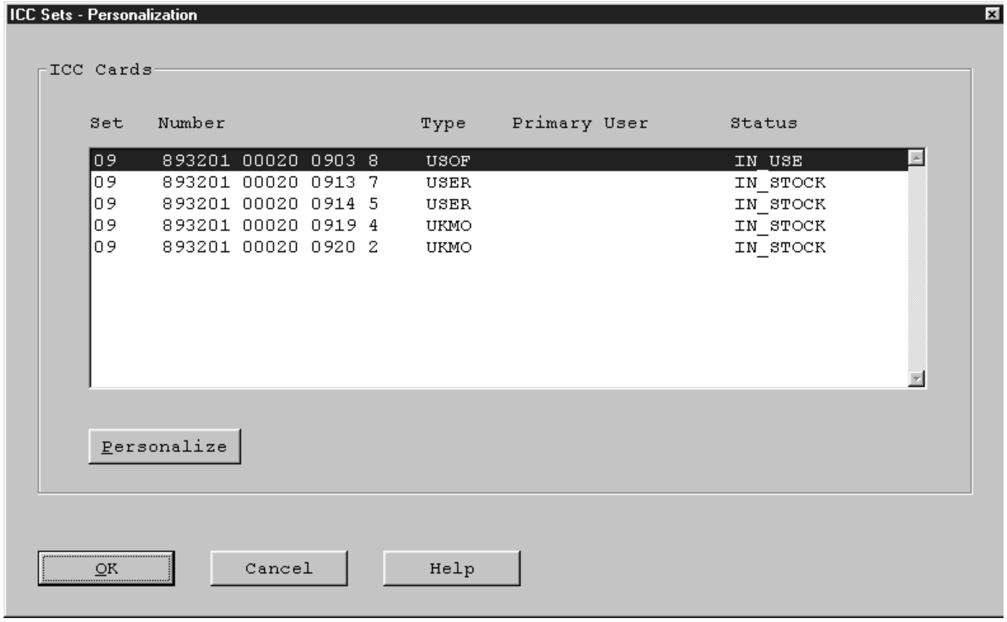
A card set always contains at least one USOF, UKMO, and USER card. You must personalize an USOF card first, because it is needed to personalize the others. Thereafter, the cards can be personalized in any order.

Note: UKMO and USER cards cannot be used until the corresponding card set is activated. Before you issue a card, check that the card belongs to the appropriate set.

To personalize an entire card set:

1. Invoke the **ICC Set Maintenance** program from the **SWIFT USE USOF** program group. The list of sets known to your system appears.
2. Select the ICC set to personalize. The status of the set must be IN STOCK.
3. Select **Personalize** from the **Selected** pull-down menu. A message window appears prompting you to insert your USOF card into the card reader.
4. Insert your USOF card into the card reader and enter your PIN or PINs, following the instructions on the card reader display.
5. Click on the **OK** push button. The PIN Parameters window appears for you to personalize the USOF card. See "Updating the PIN Parameters" on page 18 for further details.

   Note: The **Usage** parameter of a new card has a default value of 1, which means that the PIN must be changed after each usage. Select a higher value to users from having to reset PINs too often, and to prevent the card from being used up quickly.
6. Click on the **OK** push button.
7. The ICC Sets - Personalization window appears, as shown in Figure 9.



Set	Number	Type	Primary User	Status
09	893201 00020 0903 8	USOF		IN_USE
09	893201 00020 0913 7	USER		IN_STOCK
09	893201 00020 0914 5	USER		IN_STOCK
09	893201 00020 0919 4	UKMO		IN_STOCK
09	893201 00020 0920 2	UKMO		IN_STOCK

Figure 9. The ICC Sets - Personalization Window

8. Select a card to personalize, then click on the **Personalize** push button.
9. Remove the card currently in the card reader. Insert the card to be personalized and follow the instructions on the card reader display to enter the PIN or PINs. This automatically unblocks the card.
10. Click on the **OK** push button.
11. Personalize the PIN parameters for the card, as described in "Updating the PIN Parameters" on page 18, then click on the **OK** push button.
12. If the card is a USER card, the LT Access Rights window is displayed. See "Updating LT Access Rights" on page 19 for details.
13. Click on the **OK** push button to return to the ICC Sets - Personalization window.

14. When you have personalized all the cards in the set, click on the **OK** push button to return to the list of card sets.

The status of the card set changes to AVAILABLE.

Adding Existing Card Sets

Normally, you add card sets to the MERVA database using the **Order new set** and the **Acknowledge receipt** function. If, however, you want to add a set, for which the cards have already been unblocked at another system or in unconnected mode, you can use the **Read existing set** function from the **ICC Sets** pull-down menu. This function allows you to directly read the data from the cards and add it to the database.

Select the **Read existing set** function for each card of the set. You are asked to insert the card into the card reader and enter the PIN. The card is read and a window is displayed which shows the card data. For the first card of a set, you have to select the related destination from the customized list. For all additional cards, the related destination is defined through the destination ID part of the card number.

Press **OK** to add the card to the database.

After reading the cards, you use the **View/Change Details** function in the **ICC Sets and IC Cards** program to complete or change the added set and cards data. Especially the LT access rights for USER cards have to be added to the card details.

Note: Not all information available in the MERVA database for Card Sets and Cards can be read from the cards. Especially the order date cannot be retrieved and a default value will be taken by MERVA. Always use the **View/Change Details** function to check and complete the values read from the Set/Card.

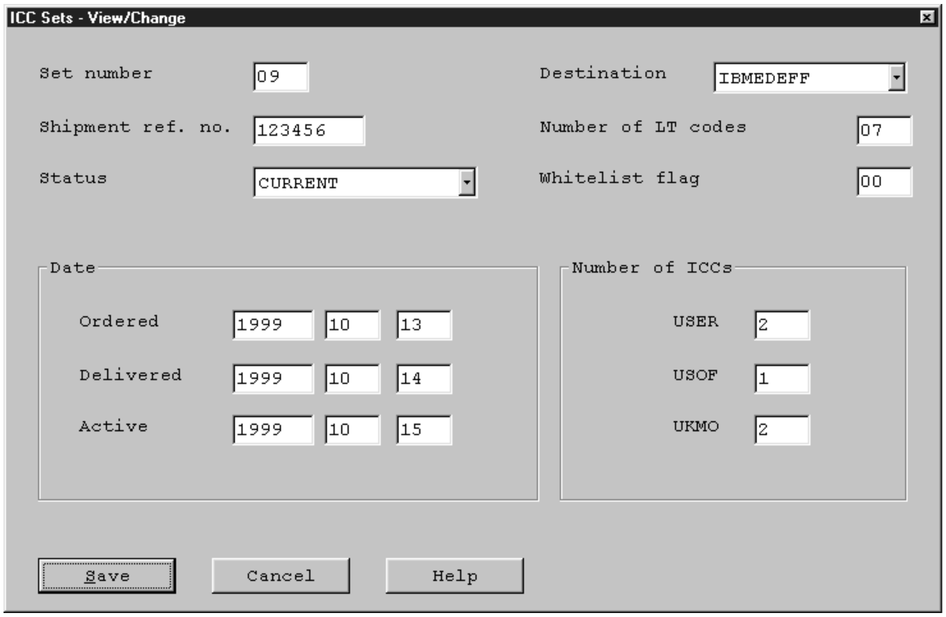
Viewing and Changing Details of a Card Set

You can view the current details of any card set. In connected mode, you can change the set parameters for the card set. In unconnected mode, you change the values that are stored in the database for a card set to ensure that they are consistent with the information stored on the cards themselves.

To view or update information about a card set:

1. Select a card set on the ICC Sets window.
2. Select **View/change details** from the **Selected** pull-down menu.

   The window shown in Figure 10 on page 17 is displayed.



Set number 09

Shipment ref. no. 123456

Status CURRENT

Destination IBMEDEFF

Number of LT codes 07

Whitelist flag 00

Date

Ordered	1999	10	13
Delivered	1999	10	14
Active	1999	10	15

Number of ICCs

USER	2
USOF	1
UKMO	2

Figure 10. The ICC Sets - View/Change Window

3. Change any of the information you want, then click on the **Save** button.

Deleting a Card Set

You must delete a card set when:

- A USOF card has been lost or is blocked and you do not have another USOF card.
- A USER card has been lost and the whitelist flag is already at the maximum possible value (30).

You can only delete a set if it is not the current set.

To delete a card set:

1. If you have already done so, activate another available set to make the card set PREVIOUS. See "Acknowledging the Receipt of a Card Set" on page 12.
2. Invoke the **ICC Set Maintenance** program from the **SWIFT USE USOF** program group. The list of sets known to your system appears.
3. Click on the entry for the set you want to delete (for example, one whose status is OBSOLETE).
4. Select **Delete** from the **Selected** pull-down menu to delete the set.
5. You are asked to confirm the deletion. Click on the **Yes** push button to delete the card set. You are asked to confirm the creation of an MT090 to send to S.W.I.F.T. with a request to delete the set. The set is then removed from the list of sets known to your system and from the database.

Administering Individual Cards

Before starting to manage ICCs, you must have at least one card reader and have defined at least one card set (using the **Card Reader Maintenance** and the **ICC Set Maintenance** program).

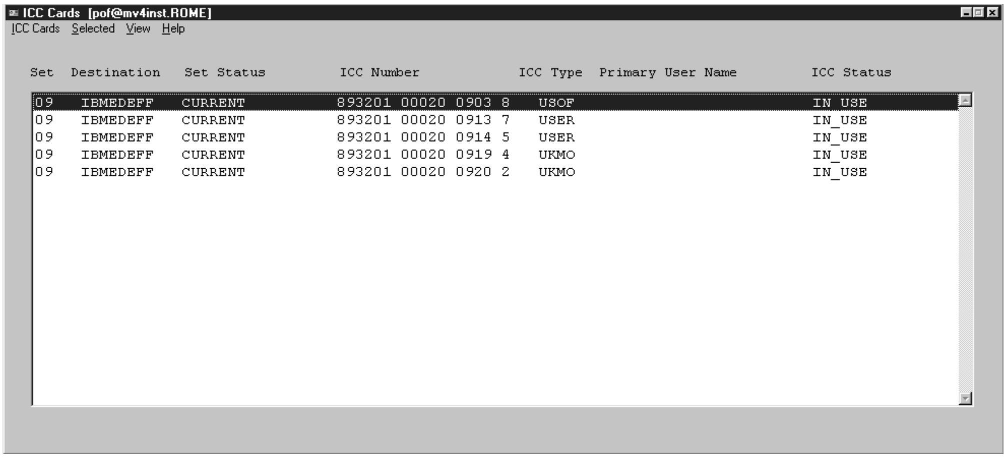
Unblocking a Card

A card can be blocked for either of the following reasons:

- When first delivered by SWIFT, all USER and UKMO cards are blocked, and must be unblocked before they can be used. However, this happens automatically when the card is personalized (see “Personalizing New Cards” on page 14).
- USER and UKMO cards become blocked when an incorrect PIN is entered three times. A card that has become blocked in this way must be unblocked before it can be used again. To unblock such a card, the card holder must know the current PIN; otherwise the card becomes void and you must destroy it.

To unblock a card:

1. Invoke the **ICC Card Maintenance** program from the **SWIFT USE USOF** program group.
2. If more than one card reader has been customized as belonging to this workstation, the Card Reader Selection window is displayed. Select the card reader to use and click on **OK**.
3. A list of cards belonging to all card sets is displayed, as shown in Figure 11.



Set	Destination	Set Status	ICC Number	ICC Type	Primary User Name	ICC Status
09	IBMEDEFF	CURRENT	893201 00020 0903 8	USOF		IN USE
09	IBMEDEFF	CURRENT	893201 00020 0913 7	USER		IN_USE
09	IBMEDEFF	CURRENT	893201 00020 0914 5	USER		IN_USE
09	IBMEDEFF	CURRENT	893201 00020 0919 4	UKMO		IN_USE
09	IBMEDEFF	CURRENT	893201 00020 0920 2	UKMO		IN_USE

Figure 11. The ICC Cards Window

4. Select the card to unblock from the list of cards on the ICC Cards window.
5. Select **Unblock** from the **Selected** pull-down menu.
6. Insert your USOF card into the card reader and enter the PIN or PINs as instructed. Click on **#** on the card reader, until the CONNECTED MODE - NOT READY message is displayed.
7. To continue, click on **YES** in the message window.
8. Insert the specified blocked card into the card reader and click on **OK**. Ask the owner of the card to enter the PIN or PINs as requested on the card reader display.
9. Click on **OK** to unblock the card.
10. Return to the **ICC Card Maintenance** program.

Updating the PIN Parameters

The card reader prompts you to change the PIN whenever it detects that the PIN is due to be updated. The card reader uses the **Usage**, **Time**, and **SK Number** PIN parameters, the current values of which are stored on the card itself, to determine if and when the PIN or PINs need to be updated.

Update the PIN parameters for a card as follows:

1. Invoke the **ICC Card Maintenance** program from the **SWIFT USE USOF** program group.
2. If more than one card reader has been customized as belonging to this workstation, the Card Reader Selection window is displayed. Select the card reader to use and click on **OK**.
3. A list of cards belonging to all card sets is displayed, as shown in Figure 11 on page 18.
4. From the list of cards on the ICC Cards window, select the card for which you want to update the PIN parameters.
5. Select **Set PIN Parameters** from the **Selected** pull-down menu.
6. Insert your USOF card into the card reader and enter the PIN or PINs as instructed. Respond to any other instructions that appear on the card reader display, until the CONNECTED MODE - NOT READY message is displayed.
7. Remove the USOF card from the card reader and insert the card to be updated.
8. In the **User 1** and **User 2** fields, enter the names of the card bearer or bearers.
9. With the **Single bearer** or **Dual bearer** radio buttons, select whether one or two users control the card.
10. In the **Usage** field, specify the number of times that a PIN can be entered correctly before it must be changed.

    The initial value displayed is that read from the card itself.
11. In the **Time** field, define the period of time (in minutes) that the card can be used before the PIN must be entered again.

    The initial value displayed is that read from the card itself.

    A value of zero means that no timeout period is set, that is, the same PIN can be used indefinitely.
12. In the **SK number** field, define the number of session keys that can be generated by a USER card before the user must enter the PIN again.

    The initial value displayed is that read from the card itself.

    A value of zero means that no limit is set, that is, any number of session keys can be generated using this PIN.

The remaining fields on the window show the number of parameter updates, PIN updates, PIN entries, and session keys remaining for this card. You can only update the PIN parameters seven times, and the card holder can only update the PIN 30 times before the card becomes permanently blocked and you must replace the card.

Updating LT Access Rights

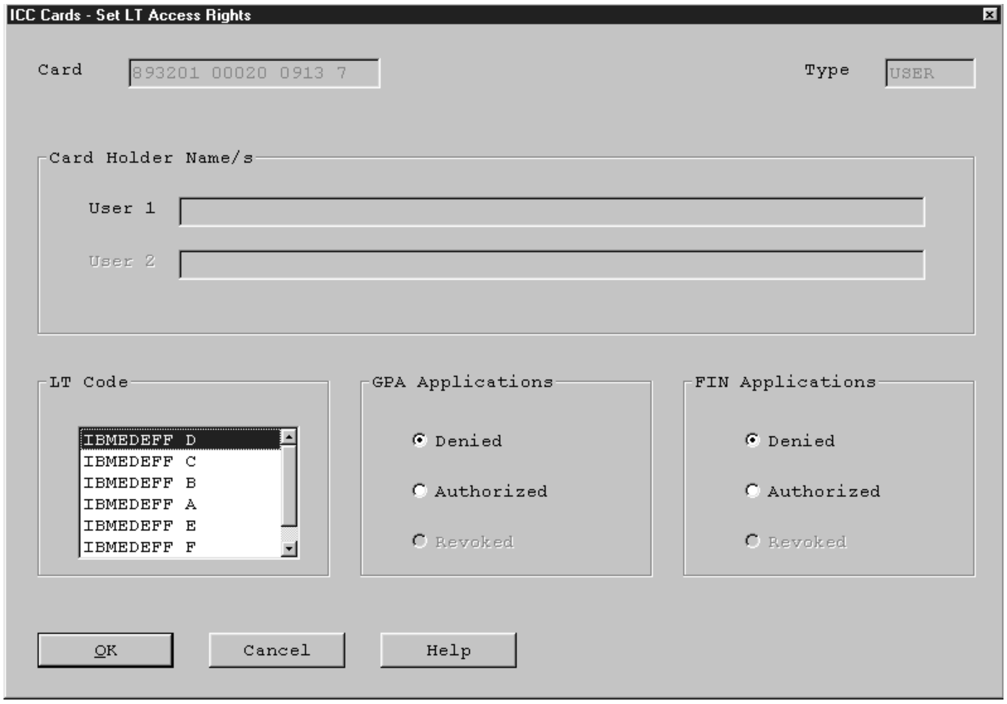
When you order a set of cards from SWIFT, the USER cards in the set contain information about the logical terminals that the cards in the set are authorized to access. When delivered, all logical terminal definitions held on the USER cards in the set are blocked to prevent unauthorized LOGIN or SELECT requests.

Before you issue each USER card, therefore, you must specify the logical terminals to which an individual USER card has access rights, and the applications (FIN or GPA) to which access is permitted.

To authorize access to logical terminals:

1. Invoke the **ICC Card Maintenance** program from the **SWIFT USE USOF** program group.
2. If more than one card reader has been customized as belonging to this workstation, the Card Reader Selection window is displayed. Select the card reader to use and click on **OK**.
3. A list of cards belonging to all card sets is displayed, as shown in Figure 11 on page 18.
4. From the list of cards on the ICC Cards window, select the USER card for which you want to update the LT access rights.
5. Select **Set LT access rights** from the **Selected** pull-down menu.
6. Insert your USOF card into the card reader and enter the PIN or PINs as instructed. Respond to any other instructions that appear on the card reader display until the CONNECTED MODE - NOT READY message is displayed.
7. Select **OK** on the message window.
8. Insert the USER card into the card reader to set the LT access rights.
9. Select **OK** on the message window.

   The ICC Cards - Set LT Access Rights window appears, as shown in Figure 12.



ICC Cards - Set LT Access Rights

Card: 893201 00020 0913 7

Type: USER

Card Holder Name/s

User 1:

User 2:

LT Code:

IBMEDEFF D
IBMEDEFF C
IBMEDEFF B
IBMEDEFF A
IBMEDEFF E
IBMEDEFF F

GPA Applications: ● Denied ○ Authorized ○ Revoked

FIN Applications: ● Denied ○ Authorized ○ Revoked

Figure 12. The ICC Cards - Set LT Access Rights Window

10. The logical terminals are listed in the **LT Code** list box. Select the logical terminal to which you want to grant access.
11. Specify the access rights to **GPA Applications** or **FIN Applications**, or both, by clicking on the **Denied**, **Authorized**, or **Revoked** radio buttons, as appropriate:

    Denied The currently selected LT cannot access the FIN or GPA application.

Authorized The currently selected LT is authorized to access the FIN or GPA application.

Revoked The currently selected LT is no longer authorized to access the FIN or GPA application.

You can only change the LT access rights for a user once. For example, if you change the access right from **Denied** to **Authorized**, you cannot subsequently change the setting back to **Denied**.

12. Repeat the previous two steps for each logical terminal to which this card is to be granted access.
13. Click on the **OK** push button to return to the list of cards.

Reading Card Data

During the procedures of personalizing a card, specifying LT access rights for a USER card, or setting the PIN parameters, MERVA reads information from the ICC. You can view this information, together with other details of an individual ICC, using the ICC Cards program. To do this, select **View/change details** from the **Selected** pull-down menu.

The following information is read from the card:

- The **Card Number**
- The card **Type** (USER, USOF, or UKMO)
- Whether the ICC is live, or is for test and training purposes only
- Whether the card has **Single bearer** or **Dual bearer** control
- The value of the PIN **Usage** parameter
- The value of the PIN **Time** parameter
- The value of the PIN **SK number** parameter.

Updating the Whitelist Flag

The whitelist flag is used to check that all the USER cards in a set are valid.

Whenever a USER card is lost from an active card set, you increment the value of the whitelist flag for the set and for all remaining user cards of the set and send an MT090 (Update Whitelist Flag Request) message informing SWIFT of the new whitelist flag value. If an attempt is then made to log in using the lost card, the SWIFT network rejects LOGIN or SELECT attempts and returns the expected value of the whitelist flag.

To ensure that the same value of whitelist flag is stored on all USER cards, the **ICC Set Maintenance** program lets you update the whitelist flag for individual USER cards in the set immediately after you have updated the value for the set.

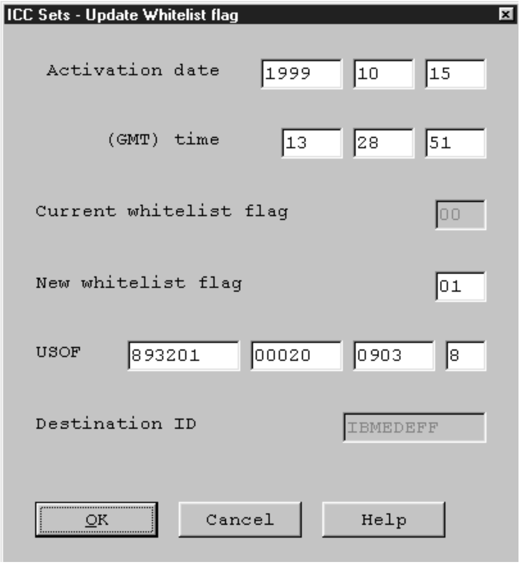
The value of the whitelist flag for individual cards can also be maintained using the **ICC Card Maintenance** program (see “Loss of a USER Card” on page 25). It is your responsibility to ensure that the value of the whitelist flags on the USER cards in the set are the same as the value for the set.

To increment the whitelist flag for the card set:

1. Invoke the **ICC Set Maintenance** program from the **SWIFT USE USOF** program group. The list of card sets known to your system appears.
2. Select the card set for which the whitelist flag is to be updated.

3. Select **Update whitelist flag** from the **Selected** pull-down menu.

   The ICC Sets - Update Whitelist Flag window appears, as shown in Figure 13.



ICC Sets - Update Whitelist flag

Activation date 1999 10 15

(GMT) time 13 28 51

Current whitelist flag 00

New whitelist flag 01

USOF 893201 00020 0903 8

Destination ID IBMEDEFF

Figure 13. The ICC Sets - Update Whitelist Flag Window

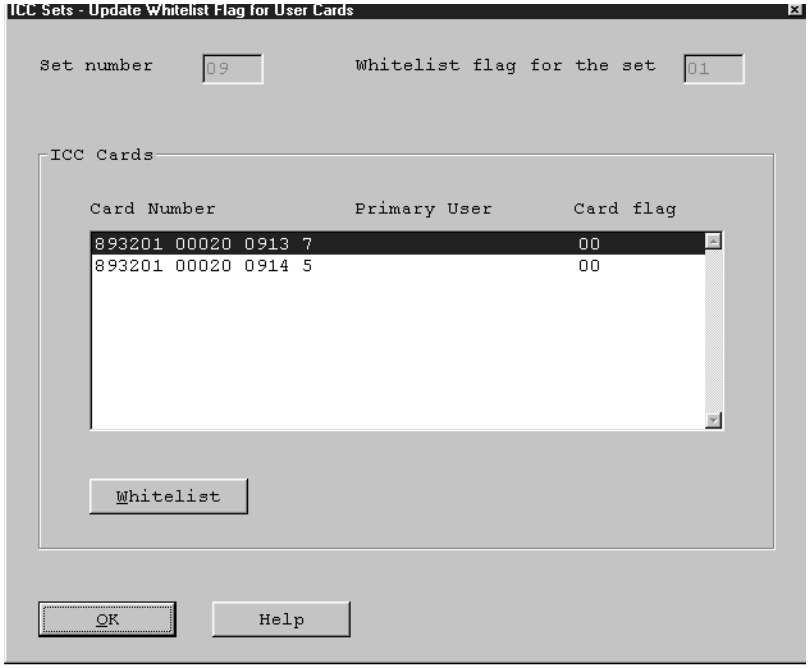
4. Specify the date and time that the whitelist flag is to be updated.
5. Specify the **New whitelist flag** value.

   This is by default one greater than the **Current whitelist flag** value displayed.
6. Check that the **USOF** card number and **Destination ID** are correct.
7. Select **OK**.
8. You are asked to confirm that you want to create an MT090 (Update Whitelist Flag Request) message and send it to SWIFT

   Select **Yes** to create the message and queue it for sending to SWIFT

   If you select **No**, you must send the message to SWIFT yourself.
9. Insert your USOF card into the card reader and enter your PIN or PINs as instructed.

   The Update Whitelist Flag for User Cards window appears, as shown in Figure 14 on page 23.



Set number 09 Whitelist flag for the set 01

ICC Cards

Card Number	Primary User	Card flag
893201 00020 0913 7		00
893201 00020 0914 5		00

Figure 14. The Update Whitelist Flag for User Cards Window

10. Select a card from the list and click on the **Whitelist** push button.
11. Update the whitelist flag for the USER card and click on the **OK** push button.
12. Repeat this procedure for all USER cards in the set.

After you have incremented the whitelist flag 29 times, the set becomes OBSOLETE and must be replaced. You replace the set by activating a new available set. The obsolete set can then be deleted, as described in “Deleting a Card Set” on page 17.

Ordering Additional Cards for an Existing Set

You order additional cards for an existing card set using the form supplied by SWIFT

When the cards have been delivered, you can record information about the cards in the database:

1. Invoke the **ICC Card Maintenance** program from the **SWIFT USE USOF** program group.
2. If more than one card reader has been customized as belonging to this workstation, the Card Reader Selection window is displayed. Select the card reader to use and click on **OK**.
3. A list of cards belonging to all card sets is displayed, as shown in Figure 11 on page 18.
4. From the ICC Cards window, select **Add cards** from the **ICC Cards** pull-down menu.
5. Select the card set’s SWIFT destination from the customized list of destinations.
6. Enter the number of the card. The set number that this card belongs to is displayed.
7. Select the card type (USER, USOF, or UKMO).
8. Click on the **Add** push button.

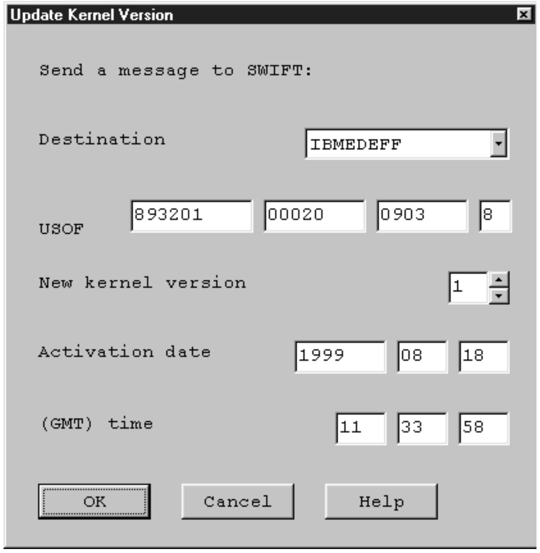
Updating the Kernel Version

An ICC kernel is a parameter used during the calculation of access codes. Eight ICC kernels are defined for each LT and are stored on all USER cards of a set. The kernel version determines which ICC kernel is currently in use.

When you update the kernel version, a message is sent to S.W.I.F.T. specifying the new kernel version number and the date and time that the change becomes effective. After the activation date and time, you must remember to specify the new kernel version on the appropriate LOGIN or SELECT panel.

To update the kernel version in use:

1. Select **Update kernel version** from the **SLS** pull-down menu.
2. The Update Kernel Version window appears, as shown in Figure 15.



Update Kernel Version

Send a message to SWIFT:

Destination: IBMEDEFF

USOF: 893201 00020 0903 8

New kernel version: 1

Activation date: 1999 08 18

(GMT) time: 11 33 58

Figure 15. The Update Kernel Version Window

3. Select the destination for the kernel version.
4. If the correct number is not already displayed, enter the **USOF** card number.
5. Specify the **New kernel version** value.
6. Specify the date and time from when the new kernel version is to be used.
7. Click on the **OK** push button.

It is your responsibility to ensure that, from the activation date and time you specify, the correct kernel version number is entered on the appropriate LOGIN and SELECT panel.

The kernel version is updated for each destination. When you update the kernel version for one LT of a destination, the kernel version used for all other LTs of that destination is updated automatically.

Invalidating Lost Cards

You must invalidate a card if it is mislaid or malfunctions.

If a card malfunctions and you still have the card, physically dispose of the card and delete the card from the set. See “Deleting a Card” on page 27. Replace the card with an equivalent from the current set. If you do not have a replacement available, either order new cards or replace the set.

If a card is mislaid, the way in which you invalidate the card varies depending on whether it is a USER, UKMO, or USOF card.

Loss of a USER Card

If you are no longer in possession of a USER card, you must increment the whitelist flag of the set and of all other USER cards in the set.

Normally, you do this by using the **ICC Set Maintenance** program from the **SWIFT USE USOF** program group (see "Updating the Whitelist Flag" on page 21). You can, however, use the **ICC Card Maintenance** program to update the whitelist flag of individual USER cards afterwards.

To increment the whitelist flag for a USER card:

1. Invoke the **ICC Card Maintenance** program from the **SWIFT USE USOF** program group.
2. If more than one card reader has been customized as belonging to this workstation, the Card Reader Selection window is displayed. Select the card reader to use and click on **OK**.
3. A list of cards belonging to all card sets is displayed, as shown in Figure 11 on page 18.
4. Select the card to update the whitelist flag for on the ICC Cards window. The card must be of the type USER and have the status IN USE.
5. Select **Update whitelist flag** from the **Selected** pull-down menu.
6. Insert your USOF card into the card reader and enter the PIN or PINs as instructed. Respond to any other instructions that appear on the card reader display until the CONNECTED MODE - NOT READY message is displayed.
7. Remove the USOF card from the card reader and insert the USER card, then select **OK**.
8. On the ICC Cards - Update Whitelist Flag window, check that the **Card Number**, **Type**, **User 1**, and **User 2** information is correct.
9. In the **New value for this card** field, specify the new value of the whitelist flag.

   By default, the value is one greater than the value specified in the **Current value for this card** field.
10. Click on the **OK** push button.

You use the **ICC Set Maintenance** program to update the whitelist flag for the set and for all USER cards in the card and then to send a message to SWIFT that the whitelist flag has been updated. If you have updated the whitelist flag for a USER card separately afterwards, the value shown in the **Current Value For This Set** field should be equal to the new value you enter in the **Current Value For This Card** field.

If you have not yet updated the set's whitelist flag, the value shown in the **Current Value For This Set** field should now be one less than the value you enter in the **Current Value For This Card** field.

To replace the lost USER card, either issue another USER card from the same set to the user, or order a replacement for the card as described in "Ordering Additional Cards for an Existing Set" on page 23.

Loss of a UKMO Card

If the UKMO card is lost, you must blacklist it. To do this:

1. Invoke the **ICC Card Maintenance** program from the **SWIFT USE USOF** program group.
2. If more than one card reader has been customized as belonging to this workstation, the Card Reader Selection window is displayed. Select the card reader to use and click on **OK**.
3. A list of cards belonging to all card sets is displayed, as shown in Figure 11 on page 18.
4. Select the card to be blacklisted from the list on the ICC Cards window. The card type must be UKMO and the card's status must be IN USE.
5. From the **Selected** pull-down menu, select **Blacklist**.
6. A message appears asking you to confirm that you want to blacklist the specified card. Select **Yes** to continue.
7. You are asked whether you want to create an MT090 (Blacklist UKMO Card Request) message and send it to SWIFT

   Select **Yes** to create the message and queue it for sending to SWIFT

   If you select **No**, you must send the message to SWIFT yourself.

The UKMO card is now blacklisted and can no longer be used. To continue, do one of the following:

- Use another UKMO card from the card set.
- Order a replacement UKMO card (see "Ordering Additional Cards for an Existing Set" on page 23).
- Activate an available card set and use the UKMO card from that set.

Loss of a USOF Card

If a USOF card is lost, you must delete the set controlled by the lost USOF card. If the set is the current set, you must first activate an available set. For details of deleting a set, see "Deleting a Card Set" on page 17.

Viewing and Changing a Card's Details

You can view the current details of any card. In connected mode, you can change the parameters for the card. In unconnected mode, you change the values that are stored in the database for a card to ensure that they are consistent with the information stored on the card itself.

To view or change details of a card:

1. Invoke the **ICC Card Maintenance** program from the **SWIFT USE USOF** program group.
2. If more than one card reader has been customized as belonging to this workstation, the Card Reader Selection window is displayed. Select the card reader to use and click on **OK**.
3. A list of cards belonging to all card sets is displayed, as shown in Figure 11 on page 18.
4. From the list of cards on the ICC Cards window, select the card to maintain, and select **View/Change Details** from the **Selected** pull-down menu. The ICC Cards - View/Change window appears.
5. Change any of the information displayed, then click on the **Save** push button.

Deleting a Card

An invalidated card can be deleted. To delete a card:

1. Invoke the **ICC Card Maintenance** program from the **SWIFT USE USOF** program group.
2. If more than one card reader has been customized as belonging to this workstation, the Card Reader Selection window is displayed. Select the card reader to use and click on **OK**.
3. A list of cards belonging to all card sets is displayed, as shown in Figure 11 on page 18.
4. From the list of cards on the ICC Cards window, select the card to delete.
5. Select **Delete** from the **Selected** pull-down menu.
6. You are asked to confirm the deletion. Click on the **Yes** push button to delete the card. The entry for the card is no longer displayed.

You are responsible for physically destroying the card.

Entering the USOF Key

The USOF key is automatically erased from the SCR's memory when one of the following events occurs:

- The default time-out period, which is specified in the *S.W.I.F.T. Security Features Technical*, expires. In this case, you are prompted to enter the USOF PIN again.
- You complete a procedure that requires the USOF PIN. This happens, for example, when you click on the **OK** push button on the LT Access Rights window.

Chapter 3. Tasks of the User Key Management Officer (UKMO)

As a user key management officer (UKMO), it is your responsibility to:

- Administer the bilateral keys which you, as a SWIFT correspondent, exchange with other SWIFT correspondents to authenticate financial messages in transit
- Administer the public key, certificate, and secure transmission key required to protect your bilateral keys
- Sett up the pre-agreement which regulates key exchange
- Deal with proposed key exchanges that have not been regulated by a pre-agreement

This involves the following tasks:

- **Administering public and secret keys**

  RSA[1] keys protect new bilateral keys while they are being exchanged. The UKMO generates an RSA key, which comprises a public key and a secret key, in a secure card reader.
- **Administering secure transmission keys**

  The secure transmission key (STK) protects the bilateral keys while they are transferred between the card reader and the workstation. The STK is also used to store the bilateral key in the workstation securely. The UKMO:
  - Generates the STK in the card reader and enters it into the database
  - Installs the STK in all card readers in your installation
  - Activates the STK.
- **Administering certificates**

  Each destination requires a certificate which guarantees that the public key is genuine. The certificate is stored in a card reader. The UKMO:
  - Requests certificates from SWIFT
  - Monitors the status of certificates
  - Revokes certificates whenever a security problem arises, for example, when a card reader is stolen
  - Manages the list of blacklisted certificates.
- **Exchanging bilateral keys**

  Bilateral keys are used to authenticate messages sent between SWIFT correspondents. The UKMO:
  - Creates a record for each correspondent financial institution with which keys are to be exchanged
  - Maintains a pre-agreement with each correspondent
  - Approves pre-agreements
  - Monitors the status of the bilateral keys being exchanged
  - Discontinues a key in an emergency
  - Starts key exchange manually, if necessary
  - Distributes keys to other systems, if distribution is not automatic
- **Dealing with incoming MT960s and MT966s**

[1] Rivest, Shamir, and Adleman.

A BKE Initiation Request message (MT960) is a request to begin key exchange. A pre-agreement covering the terms of the key exchange must have been approved before an MT960 is sent. If an MT960 arrives that does not have an approved pre-agreement, the UKMO must decide whether or not to proceed with key exchange.

- **Backing up and restoring bilateral keys and pre-agreements**

  The UKMO is responsible for regularly backing up and restoring bilateral keys and pre-agreements.

Figure 16 shows the program group for the UKMO administration tasks.

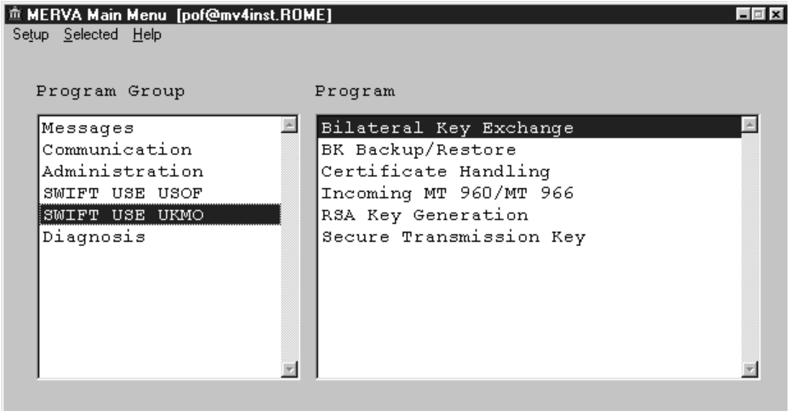


Figure 16. The User Key Management Officer Program Group

When you are preparing for bilateral key exchange, work through the programs in the order shown in the following table, beginning with the **RSA Key Generation** program. This enables you to complete the administration tasks in the correct order.

Program Name	**Administration Function**
RSA Key Generation	Administering public keys, see "Generating Public and Secret Keys" on page 31.
Secure Transmission Key	Administering secure transmission keys, see "Administering Secure Transmission Keys" on page 32.
Certificate Handling	Administering certificates, see "Managing Certificate Values (CVs)" on page 39.
Bilateral Key Exchange	Exchanging bilateral keys, see "Exchanging Bilateral Keys" on page 46.
Incoming MT960/MT966	Deal with incoming MT960 (BKE Initiation Request) and MT966 (BKE Discontinuation Request) messages that are not automatically processed. See "Dealing with Incoming MT960 and MT966s" on page 70.
BK Backup/Restore	Make a copy of your bilateral keys to a location you choose (see "Backing Up Bilateral Keys and Pre-Agreements" on page 73), or restore your copy of the bilateral keys, overwriting the existing versions stored by MERVA (see "Restoring Bilateral Keys and Pre-Agreements" on page 78).

Upgraded Secure Card Readers

S.W.I.F.T. has introduced a new type of secure card reader (SCR) that has a label with the text **Upgraded SCR** on it. The upgraded SCR is fully compatible with the old SCR. If the software is not updated, then all functions work as before, but the new functions are not available. For more information, see "Appendix A. Upgraded Secure Card Reader" on page 93.

Prerequisites for BKE

You must have a public key, a certificate, a secure transmission key, and an installed certificate blacklist before you can exchange bilateral keys.

Generating Public and Secret Keys

The process of enciphering bilateral keys relies on the use of two encryption keys, the *public key* and the *secret key*. These keys are known as RSA keys and are used to encipher the bilateral key while it is in transit over the SWIFT network in such a way that only the intended recipient can decipher it.

Both keys are generated inside the SCR. The public key is sent to correspondents with the message, while the secret key always remains inside the SCR.

Your public key enciphers a bilateral key in transit from a correspondent to you. Similarly, your correspondent's public key enciphers a bilateral key while it is in transit from you to the correspondent.

Bilateral keys enciphered using a public key can only be deciphered in the appropriate secure card reader by the matching secret key.

You must always generate new RSA keys in accordance with SWIFT recommendations. Generate new RSA keys:

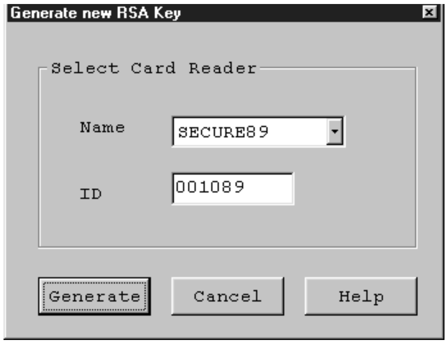
- When you first start using the BKE service
- If you suspect that the security of the secure card reader where the secret key is stored has been compromised

For each destination, you must apply for a certificate from SWIFT that the public key is genuine. When you have generated a public key, therefore, request a certificate for it using the Certificate Handling program.

To generate a public key:

1. Invoke the **RSA Key Generation** program from the **SWIFT USE UKMO** program group.
2. Select **Generate** from the **RSA Key** pull-down menu.

   The Generate New RSA Key window appears, as shown in Figure 17 on page 32.



Name SECURE89

ID 001089

Figure 17. The Generate New RSA Key Window

3. Select the **Name** of the card reader in which to generate the RSA keys from the drop-down list box. The serial number (**ID**) of the card reader is then displayed.
4. Click on the **Generate** push button.
5. A message appears warning you that any certificates held for the existing public key are invalidated when a new key is generated.

   Click on **YES** to continue.

   Note: The status of existing certificates is changed to ARCHIVED. If you interrupt the RSA change process on the card reader, MERVA does not get a notice and the status is changed nevertheless. In this case, to get the correct certificate information, delete the certificates from the database and read the certificate value (CV) IDs from the card reader using the function **Read CV-IDs from SCR**.
6. Insert your UKMO card into the card reader and click on the **OK** push button.
7. Follow the instructions on the card reader display to generate the RSA keys.

Administering Secure Transmission Keys

The secure transmission key (STK) is used to encrypt all your bilateral keys. Bilateral keys are generated in a secure card reader and then transferred to the workstation for key exchange and subsequent use. Before a bilateral key leaves the SCR, it is encrypted using the STK to ensure the security of the key. The STK is stored in an encrypted form on the workstation.

Notes:

1. When you generate a new STK, all existing bilateral keys must be re-enciphered using the new STK. Therefore, before you generate the new STK, you must back up your bilateral keys. If you follow the steps described in "Generating a Secure Transmission Key" on page 33, you are prompted to do a backup of your bilateral keys. You can also do a backup of all keys in MERVA format with the BK Backup/Restore program before starting the Secure Transmission Key program. See "Backing Up Bilateral Keys and Pre-Agreements" on page 73 for details of how to do this.
2. Before you can begin to exchange bilateral keys, you must generate an STK in a secure card reader, then enter the values representing the STK from the card reader into your workstation. See Figure 18 on page 34 for details.
3. You are then prompted to activate the STK in the secure card reader, where you generated it.

4. After the STK has been generated in one secure card reader, entered into the database on your workstation, and activated on the secure card reader, you are prompted to restore the previously backed-up bilateral keys. During restore the keys are automatically encrypted using the new STK. See "Restoring Bilateral Keys and Pre-Agreements" on page 78 for details.
5. If you have more than one secure card reader, you must install and activate the new STK in all other card readers in your installation. See Figure 21 on page 38 for details.
6. If the STK was not activated during the STK generation process, you can subsequently activate the STK separately. See Figure 22 on page 39 for details of how to do this.
7. If you are not sure whether you have restored your bilateral keys after the STK was entered into the database, use the BK Backup/Restore program as described in "Restoring Bilateral Keys and Pre-Agreements" on page 78 to restore your bilateral keys.

   If you have made a backup of your bilateral keys before generating the new STK, there is **no problem** in restoring the backup file twice.
8. To ensure data integrity, you cannot use any other MERVA program in parallel to the STK program. Logout from SWIFT and logoff all other workstations. After changing the STK logoff from MERVA and logon again. Otherwise the USE background process does not know that the STK has changed.

Generating a Secure Transmission Key

Generate the STK:

- In accordance with SWIFT recommendations (for example, every six months if your installation deals with a high volume of messages).
- Whenever a secure card reader is broken or mislaid.
- If you mislay your safe copy of the STK.

To minimize the security risk, an STK is never transmitted over the physical connection between the workstation and the card reader, even if the card reader is operating in connected mode. You must read the data from the card reader and enter it into the workstation manually.

Notes:

1. The procedure to follow for MERVA ESA is described in the *MERVA ESA V4 Operations Guide* (DWSAUTLD program).
2. The STK displayed on the SCR consists of two parts. Each half can be controlled by a separate person. The following describes the procedure in which two UKMOs are involved in generating an STK.

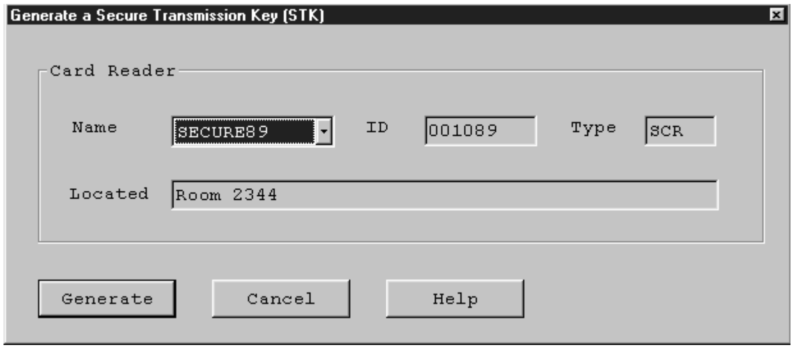
To generate an STK:

1. Invoke the **Secure Transmission Key** program from the **SWIFT USE UKMO** program group. The Secure Transmission Keys (STK) window appears.
2. Select **Generate** from the **Secure transmission key** pull-down menu.
3. You must have an up-to-date backup copy of your bilateral keys before you generate a new STK. If bilateral keys are already stored in the database, the Backup Bilateral Keys window appears.

   You may select *No backup* only if you already performed a backup before using the BK Backup/Restore program with the parameters *MERVA* format, *KMA=ALL* and *Complete File* just before. The name of the log file written during backup is **STKBACK.LOG**. This file is located in the home directory of your system.

After generation of the STK, installation in the database and activation of the STK in the card reader, you are prompted to do a restore of the backup file. The name of the log file written during restore is **STKREST.LOG**. This file is located in the home directory of your system.

4. The Generate a Secure Transmission Key window appears, as shown in Figure 18.



Name SECURE89

ID 001089

Type SCR

Located Room 2344

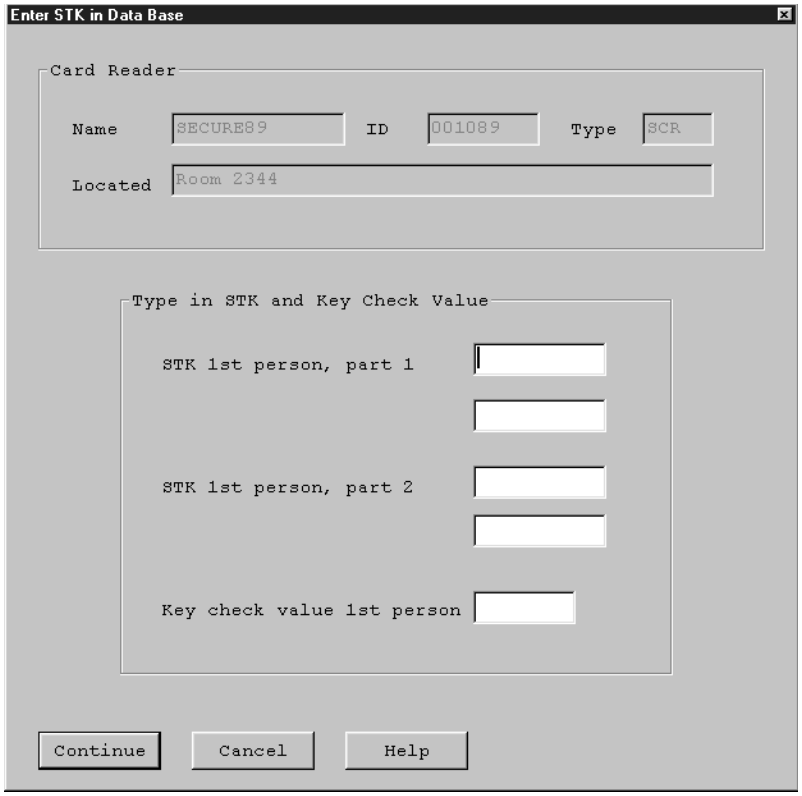
Figure 18. The Generate a Secure Transmission Key (STK) Window

Proceed as follows:

5. Select the name of the secure card reader in which you wish to generate the STK from the **Name** drop-down list. The serial number of the card reader is then displayed in the **ID** field.

   The comment in the **Located** entry field indicates the physical location of the card reader and was originally entered with the Card Reader Maintenance program.
6. Click on the **Generate** push button.
7. Insert your UKMO card into the card reader and enter your PIN or PINs as instructed.

   The Enter Secure Transmission Key (STK) in Database window is displayed, as shown in Figure 19 on page 35.



Enter STK in Data Base

Card Reader

Name SECURE89 ID 001089 Type SCR

Located Room 2344

Type in STK and Key Check Value

Figure 19. The Enter Secure Transmission Key (STK) in DB Window

8. The first half of parts 1 and 2 of the STK and the STK check value are displayed on the card reader.

> **Important note!**
>
> Keep a safe copy of these details, since this key is not stored and you may need it later, for example, when installing the STK on new card readers or after reinstalling the database to restore backed-up keys.

Notice that, on the Enter Secure Transmission Key (STK) window, the layout of the empty fields for the first half of an STK has a similar format to the card reader layout.

9. Enter the first half of the STK (STK1) and the key check value into the window.
10. Click on the **Continue** push button to enter the data.
11. A message appears asking you whether you want to print the values you have entered as input value for the first half of the STK. Click on **Yes** to print the information—the Print window is displayed and you can choose to print the information to the file or printer of your choice. Otherwise, click on **No**.
12. The second UKMO now repeats steps 2 to 4 for the second half of the STK.
13. Click on the **Continue** push button on the message window that is displayed to enter the data.
14. A message appears asking you whether you want to print the values you entered as input values for the second half of the STK. Click on **Yes** to print

the information—the Print window is displayed and you can choose to print the information to the file or printer of your choice. Otherwise, click on **No**.

15. The Activate Secure Transmission Key (STK) window is displayed.

You must activate the STK on this card reader immediately by completing the following steps (or after installation, by completing the steps in Figure 22 on page 39).

To activate the STK:

1. Select the name of the card reader from the **Name** drop-down list. The serial number of the card reader is then displayed in the **ID** field.
2. Click on the **Activate** push button.
3. The STK is now activated in the SCR, and the STK Activation Complete window appears. Click on the **OK** push button to acknowledge activation of the new STK.

Entering the STK into the Database

Note: Changing the Secure Transmission Key (STK) affects all the existing Authentication keys in the database. So it is mandatory to do a BK Backup before entering the new STK into the Database or do it by the BK Backup/Restore program in MERVA format with *KMA=ALL* and *Complete File*. After the STK has been entered into the database, do a restore of the file created by the BK Backup before.

To enter the secure transmission key into the database:

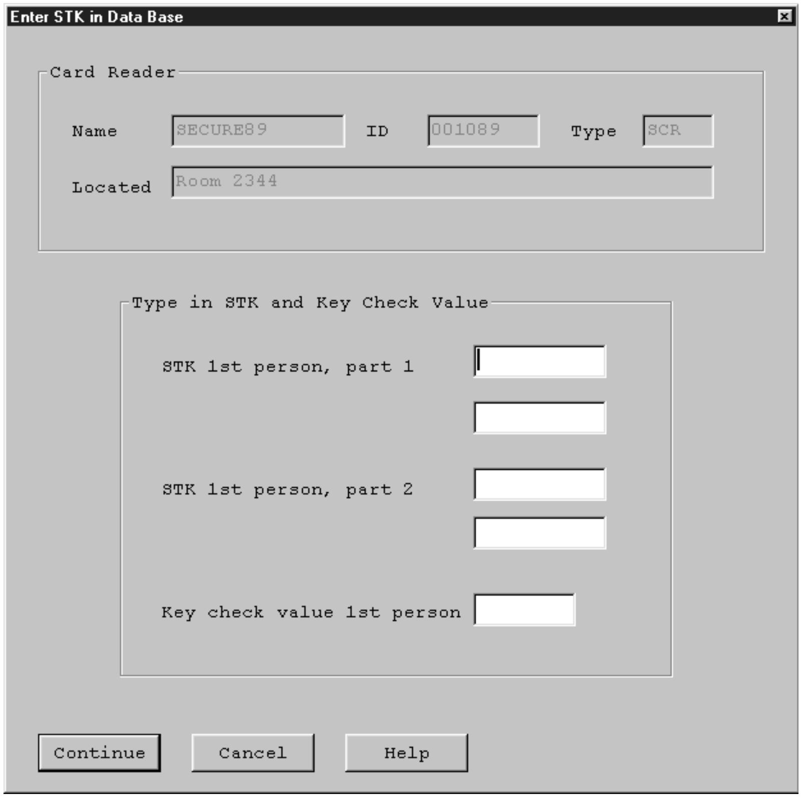
1. Select **Enter into DB** from the **Secure Transmission Key** pull-down menu.

   MERVA asks you if you want to backup the existing bilateral keys. You need a backup of the keys because changing the STK affects all the existing keys in the database.

   You may select *No backup* only if you already performed a backup before using the BK Backup/Restore program with the parameters *MERVA* format, *KMA=ALL* and *Complete File* just before. The name of the log file written during backup is **STKBACK.LOG**. This file is located in the home directory of your system.

   After you have entered the STK into the Database, do a restore of the backup file. The name of the log file written during restore is **STKREST.LOG**. This file is located in the home directory of your system.

   The Enter Secure Transmission Key (STK) in DB window is displayed, as shown in Figure 20 on page 37.



Name SECURE89 ID 001089 Type SCR

Located Room 2344

Figure 20. The Enter Secure Transmission Key (STK) in DB Window

2. Enter the first half of the STK (STK1) and the key check value into the window.
3. Click on the **Continue** push button to enter the data.
4. Enter the second half of the STK (STK2) and the key check value into the window.
5. Click on the **Continue** push button to enter the data.
6. You are asked whether you want to restore your bilateral keys now.

   Note: If you enter a new STK into the database in a separate step, not during generation of the STK, **restore the bilateral keys now**.

   After that the new STK has to be *installed* and *activated* in all other SCRs of your installation.

   You can only begin to exchange bilateral keys if the STK is in your database, all existing bilateral keys are encrypted with this STK, and the STK is installed and activated in all SCRs.

   Click on the **Restore** push button. The Restore Bilateral Keys window appears. Enter the name of the file containing the backed-up bilateral keys in the Restore Bilateral Keys window and click on the **Restore** push button. The bilateral keys are restored and automatically encrypted with the new STK you have entered into the database.

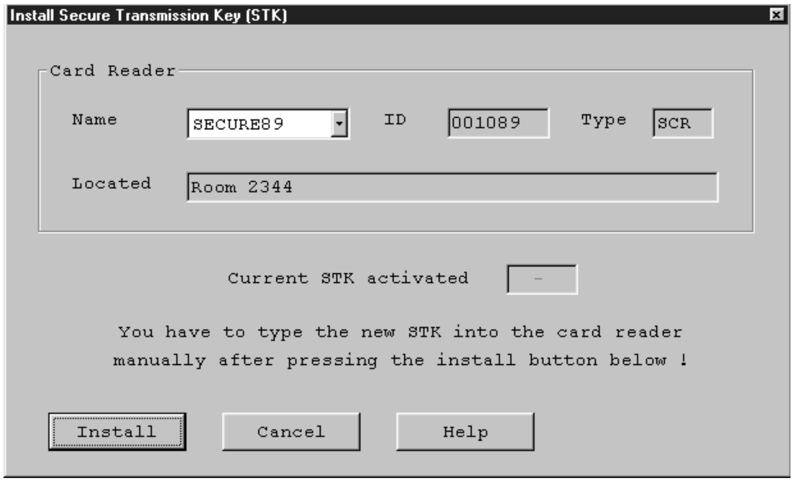
Installing a Secure Transmission Key

Only one secure transmission key is generated for an installation, even when several card readers are in use (or when a second card reader is to be kept ready for use as a backup).

Therefore, after you have generated an STK in an SCR and copied it to the workstation directly attached to the card reader, the new STK must be installed and activated in all other SCRs in your installation.

To do this, do the following:

1. Display the Secure Transmission Keys (STK) window.
2. Select **Install** from the **Secure Transmission Key** pull-down menu. The Install Secure Transmission Key window appears, as shown in Figure 21.



Name SECURE89 ID 001089 Type SCR

Located Room 2344

Current STK activated -

You have to type the new STK into the card reader manually after pressing the install button below !

Figure 21. The Install Secure Transmission Key Window

3. Select the name of the card reader in which to install the new STK from the **Name** drop-down list. The serial number of the card reader is then displayed in the **ID** field.
4. Click on the **Install** push button.
5. A message window appears prompting you to insert your UKMO card.
6. Insert your UKMO card into the secure card reader and click on **OK**.
7. Start the STK installation function on the card reader. See the *S.W.I.F.T. Card Readers User Guide* for details of how to do this.
8. The Activate Secure Transmission Key window appears. Using your safe copy of the STK, enter the first half of the STK into the card reader.
9. The window for the second half of the new STK (STK2) is displayed. Enter the second half of the STK into the card reader.

Activate the new STK on the workstation following the procedure described in Figure 22 on page 39.

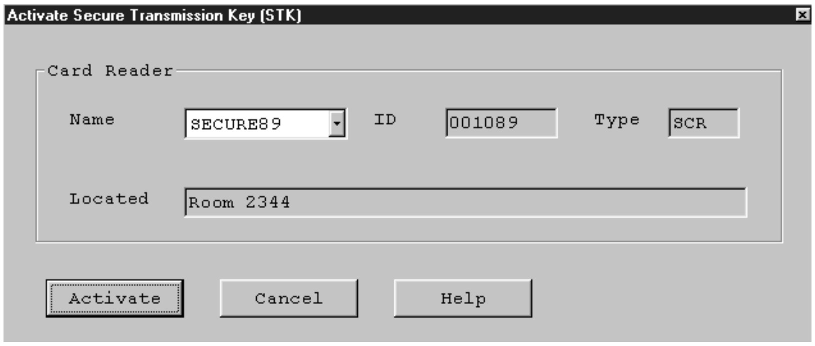
Repeat this procedure for all other secure card readers in your installation.

Activating a Secure Transmission Key

Once the STK has been generated or installed in the SCR, the STK must be activated. Activation of the new STK usually takes place during the generation stage, but can be performed separately later after installing the STK in another card reader.

To activate an STK:

1. Display the Secure Transmission Keys (STK) window.
2. Select **Activate** from the **Secure Transmission Keys** pull-down menu. The Activate Secure Transmission Key window appears, as shown in Figure 22.



Activate Secure Transmission Key (STK)

Card Reader

Name: SECURE89

ID: 001089

Type: SCR

Located: Room 2344

Figure 22. The Activate a Secure Transmission Key (STK) Window

3. Select the name of the card reader from the **Name** drop-down list. The serial number of the card reader is then displayed in the **ID** field.
4. Click on the **Activate** push button.
5. A message window appears prompting you to insert your UKMO card.
6. Insert your UKMO card into the secure card reader and click on **OK**.

   When activation is complete, the STK - Activation Complete window appears.
7. Click on the **OK** push button.
8. If the Activate Secure Transmission Key window appeared automatically after the STK generation, then the Restore Bilateral Keys window appears.
9. Enter the name of the file containing the backed-up bilateral keys and click on the **Restore** push button.

   If the bilateral keys were restored previously **after the new STK has been entered into the database**, click on the **Cancel** push button.

Managing Certificate Values (CVs)

Once you have generated a public key, you must apply for a certificate value (CV) from S.W.I.F.T. to verify that it is genuine. A CV is mandatory for each destination before bilateral key exchange can begin.

If you have an SCR that has not been upgraded, you will receive one CV, which is based on the PKS currently used by SWIFT If you have an upgraded SCR, you will receive two CVs: one based on the current PKS, and one on the previous version of the PKS. For more information about upgraded SCRs, see “Appendix A. Upgraded Secure Card Reader” on page 93.

Listing Specific CVs

You use the Certificate Handling program from the User Key Management Officer folder to manage certificates. When you start the program, a list of all existing certificates is displayed showing information about the current status of the certificates.

If you want to include only a certain type of certificate in the list of certificates, select **Include** from the **View** pull-down menu. A cascaded menu appears, from which you can select your *own* certificates, *blacklisted* certificates, or *all* certificates. If, for example, you select **Blacklisted**, only blacklisted certificates are displayed in the list.

Requesting a New CV

Request a certificate:

- In accordance with SWIFT recommendations
- When you generate a new public key
- When SWIFT changes its public key.

A certificate applies to a public key, so you must have generated a public key before you can request a certificate.

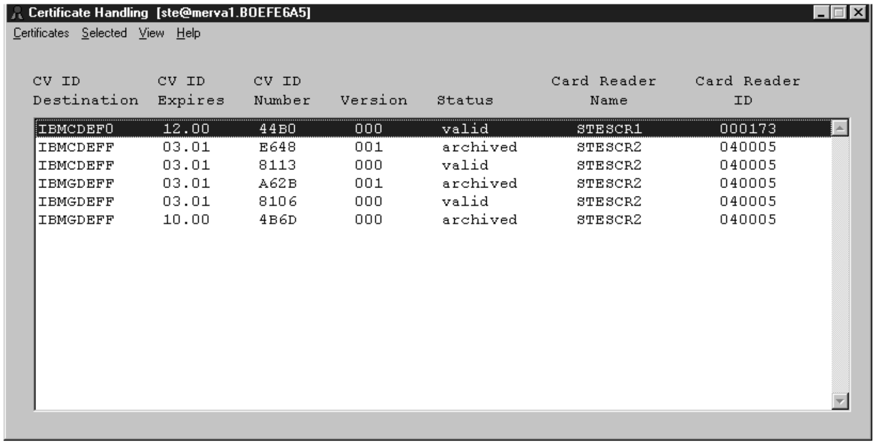
Up to 8 certificates (8 for live destination plus 8 for T&T destinations) can be stored in the old card reader. The upgraded card reader can contain one set per version. Two versions are posssible. for example:

- 8 CVs for live destinations version 0
- 8 CVs for T&T destinations version 0
- 8 CVs for live destinations version 1
- 8 CVs for T&T destinations version 1

To request a certificate:

1. Invoke the **Certificate Handling** program from the **SWIFT USE UKMO** program group. The list of certificates appears, as shown in Figure 23 on page 41. For a certificate that has been requested from S.W.I.F.T. but has not yet been delivered:
   - For an old SCR, four question marks (**????**) are displayed in place of the CV-ID number.
   - For an upgraded SCR, two versions are requested automatically, and three question marks plus the number 1 (for the lower version) or 2 (for the higher version) are displayed in place of the CV-ID number (**???1** or **???2**).

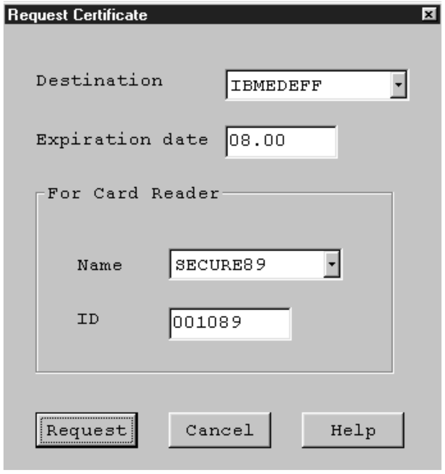
   When the certificate arrives, then the received CV ID is displayed instead of the question marks.


Certificate Handling [ste@merva1.BOEFE6A5]
Certificates Selected View Help

CV ID Destination	CV ID Expires	CV ID Number	Version	Status	Card Reader Name	Card Reader ID
IBMCDEF0	12.00	44B0	000	valid	STESCR1	000173
IBMCDEFF	03.01	E648	001	archived	STESCR2	040005
IBMCDEFF	03.01	8113	000	valid	STESCR2	040005
IBMGDEFF	03.01	A62B	001	archived	STESCR2	040005
IBMGDEFF	03.01	8106	000	valid	STESCR2	040005
IBMGDEFF	10.00	4B6D	000	archived	STESCR2	040005

Figure 23. The Certificates Window

2. From the **Certificates** pull-down menu, select **Request Certificate**. The Request Certificate window appears, as shown in Figure 24.



Destination IBMEDEFF

Expiration date 08.00

Name SECURE89

ID 001089

Figure 24. The Request Certificate Window

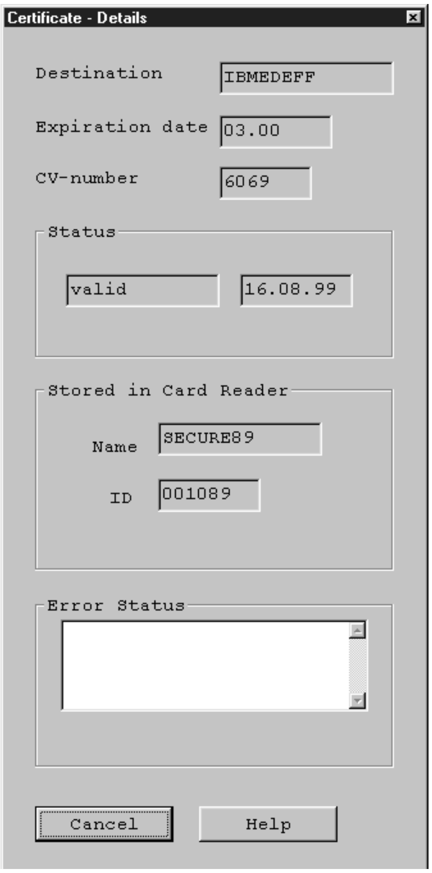
3. Select the **Destination** from the customized list of destinations.
4. Enter the **Expiration date** in the form *MM.YY*, where *MM* is the month and *YY* the year. The default is one year from now.
5. Select the **Name** of the card reader from the drop-down list. The serial number (**ID**) of the card reader is then displayed.
6. Insert your UKMO card and enter the PIN or PINs as instructed.
7. Click on the **Request** push button.

   An MT075 (Certification Request) message is generated (for an upgraded SCR, one for each version), and is queued for routing to SWIFT
8. On the Certificates window, select **Refresh now** from the **View** pull-down menu to update the list of certificates.

Displaying the Details of a CV

The details of a certificate appear on the Certificate - Details window. To display the details of a certificate:

1. Invoke the **Certificate Handling** program from the **SWIFT USE UKMO** program group. The list of certificates appears, as shown in Figure 23 on page 41.
2. Select the certificate for which you require the details.
3. Select **Open as details** from the **Selected** pull-down menu. The Certificate - Details window appears, as shown in Figure 25.



Certificate - Details

Destination: IBMEDEFF

Expiration date: 03.00

CV-number: 6069

Status: valid 16.08.99

Stored in Card Reader

Name: SECURE89

ID: 001089

Error Status:

Figure 25. The Certificate - Details Window

This window shows:

- The destination to which the certificate belongs
- The date that the certificate expires
- The number of the certificate
- The status and the date that the certificate acquired this status. The status can be one of:

  requested The certificate has been requested from SWIFT but the MT087 (certification response) message has not yet arrived.

  valid The certificate is associated with an authentic public key, and has not expired. The MT087 (certification response) message has been received.

  revoking The revoke request has been sent to SWIFT, but the acknowledgment has not yet been received.

  blacklisted The certificate is blacklisted.

archived Although this certificate is still valid, another certificate has been activated.

error An error occurred. More information is displayed in the Error Status field, but to understand what happened, you will have to check the diagnostic log.

- The name and serial number (ID) of the SCR in which the certificate is stored.
- When an error occurs (the status is set to **error**), more information about the error is displayed in the Error Status field.

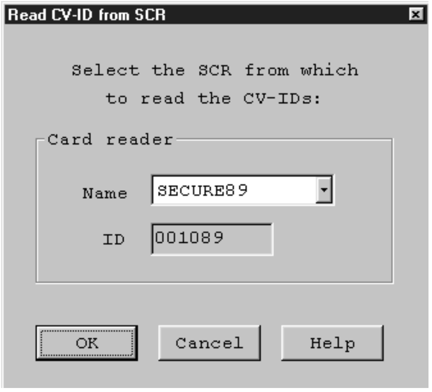
Notes:

1. For an upgraded SCR, two certicates are requested for each destination. It is possible that the one or both of the requested certificates fails. If so, delete the entries that contain the error information and request a new certificate.
2. The destination, expiration date, and number of the certificate form the certificate identity (CV-ID).
3. The Details window does not show the version of the certificate; this is shown only in the Certificates window (see Figure 23 on page 41).

Reading CV-IDs from the SCR

If you attach another SCR to your workstation, the certificate identities (CV-IDs) stored in that SCR can be uploaded to your database. To read the CV-IDs from an SCR:

1. Invoke the **Certificate Handling** program from the **SWIFT USE UKMO** program group. The list of certificates appears, as shown in Figure 23 on page 41.
2. Select **Read CV-IDs from SCR** from the **Certificates** pull-down menu. The Read CV-ID from SCR window appears, as shown in Figure 26.



Read CV-ID from SCR

Select the SCR from which to read the CV-IDs:

Card reader

Name: SECURE89

ID: 001089

Figure 26. The Read CV-ID from SCR Window

3. Select the name of the card reader from the drop-down list. The serial number (ID) of the card reader is then displayed.
4. Click on the **OK** push button to read the CV-IDs.

Revoking CVs

Since a certificate guarantees the validity of a public key, you must revoke a certificate immediately you suspect that the security of the key has been compromised.

In particular, if an SCR is stolen, **all** certificates stored in the SCR must be revoked.

You can only revoke your own destination’s certificates.

To revoke a certificate, you request SWIFT to blacklist the certificate. SWIFT issues a broadcast message to all correspondents listing the blacklisted certificate. All correspondents must refuse any public key “guaranteed” by a blacklisted certificate.

From your point of view as UKMO:

- When you initially prepare for bilateral key exchange, you must obtain the complete up-to-date certificate blacklist from SWIFT, see “Requesting the CV Blacklist”.
- Whenever you receive a broadcast message from SWIFT listing newly blacklisted certificates, the system attempts to update the list of certificates. You must check that the list is correct and, if necessary, manually change the status of the certificates in the database to BLACKLISTED.

To revoke the certificate:

1. Invoke the **Certificate Handling** program from the **SWIFT USE UKMO** program group. The list of certificates appears, as shown in Figure 23 on page 41.
2. Select the certificate you wish to revoke. It must be one of your own destination’s and be installed in an SCR.
3. Select **Revoke** from the **Selected** pull-down menu. The Confirm - Revoke Selected Certificates window appears.
4. Select your Broadcast Center from the list or enter it into the entry field.
5. Click on the **Revoke** push button.

The status of the certificate is changed to REVOKING and an MT074 (CV Revocation Broadcast Request) message sent to SWIFT

When the MT094 (CV Revocation Broadcast Message) is received, check that the status of the certificate or certificates specified in the free-format message have been changed to BLACKLISTED. If any have not, update the certificate’s status yourself using the **Enter Blacklisted CVs into DB** menu item (see “Entering Blacklisted CVs into the Database” on page 45).

Requesting the CV Blacklist

To get the complete up-to-date blacklist from SWIFT:

1. Invoke the **Certificate Handling** program from the **SWIFT USE UKMO** program group. The list of certificates appears, as shown in Figure 23 on page 41.
2. Select **Request blacklist** from the **Certificates** pull-down menu. The Confirm - Request Blacklist window appears.
3. Select the **Emitting destination**, that is, your SWIFT destination address that appears on the message sent to SWIFT as the requester of the CV blacklist.
4. Click on the **Request** push button.

When you receive a blacklist update, the system automatically updates the list of certificates.

Entering Blacklisted CVs into the Database

When the MT094 (CV Revocation Broadcast Message) is received, the system attempts to identify the CVs specified in the message and blacklist the CVs automatically. In case this fails, you must check the contents of the message and update the certificate's status yourself.

To do this:

1. Identify the CVs that have not been changed to BLACKLISTED status in the database by checking the contents of the MT094 message against the Status column of the certificates window.
2. Select **Enter Blacklisted CVs into DB** from the **Certificates** pull-down menu.

   The Enter Blacklisted CV into Database window is displayed, as shown in Figure 27.

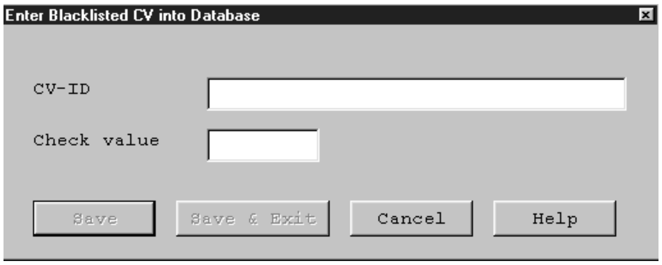


Figure 27. The Enter Blacklisted CV into Database Window

3. Enter the **CV-ID** of a certificate and the 4-character **Check value** for the CV-ID. You can find these values in the free-format text of the MT094 (CV Revocation Broadcast) message.
4. If you are entering only one certificate, click on **Save and Exit**. The status of the certificate in the database is changed to BLACKLISTED, and you are returned to the Certificates window.

   If you are entering more than one CV, click on **Save**, then continue to enter the information for the remaining CVs.

Note: You must have selected **Include Blacklisted** from the **Edit** pull-down menu for the blacklisted certificates to be listed on the Certificates window.

Deleting a Certificate

A certificate is deleted when it has been archived and its expiration date has passed.The USE background process checks for expired CVs each day (at next startup, or at midnight), and deletes them from the database. To keep the SCR's table from becoming full, delete expired and archived CVs from the SCR. It is your responsibility as UKMO to avoid deleting valid certificates.

To delete the certificate from the SCR and database:

1. Invoke the **Certificate Handling** program from the **SWIFT USE UKMO** program group. The list of certificates appears, as shown in Figure 23 on page 41.
2. Select **Read CV-IDs from SCR** from the certificates pull down menu. After selecting the appropriate card reader, all CV-IDs stored in the card reader are read and then displayed.
3. Select the CVs with status ARCHIVED or EXPIRED, and select **Delete** from the selected pull down menu.

4. The Confirm - Delete CV(s) from SCR window is displayed. You want to delete the CVs from both the database and the SCR, so press the **Delete** button; do not click on ″Delete CV(s) only from database″.
5. Insert the UKMO card as requested and enter the PIN or PINs as requested.
6. Press the OK button. The certificates are deleted from the card reader and from the database, but are not blacklisted by SWIFT

If expired certificates are stored in the SCR, and the table for certificates is full, then it is may not possible to delete these certificates from MERVA because these certificates have already been deleted from the database. Switch the SCT to unconnected mode, insert the UKMO card, and follow the instruction of card readers guide to delete the expired CVs. You can also use the function **Read CVs from RDR** to store the expired CVs into the database. After invoking this function, you can see the expired CVs on the display, and you can select the CVs for deletion. At midnight or the next time MERVA is started, the USE batch process will erase all these expired CVs from the database.

Exchanging Bilateral Keys

Before being able to exchange bilateral keys with a correspondent, you must have:

- Generated a public key
- Obtained a certificate that the public key is genuine
- Generated a secure transmission key
- Installed a certificate blacklist.

The next step is to reach an agreement (termed a pre-agreement) with your correspondent. This pre-agreement specifies the terms and conditions under which the keys are exchanged, such as the type of key to use and the length of time that the key remains valid. Details about the terms of the pre-agreement are stored in the database. Once terms have been agreed, key exchange takes place automatically at the agreed time.

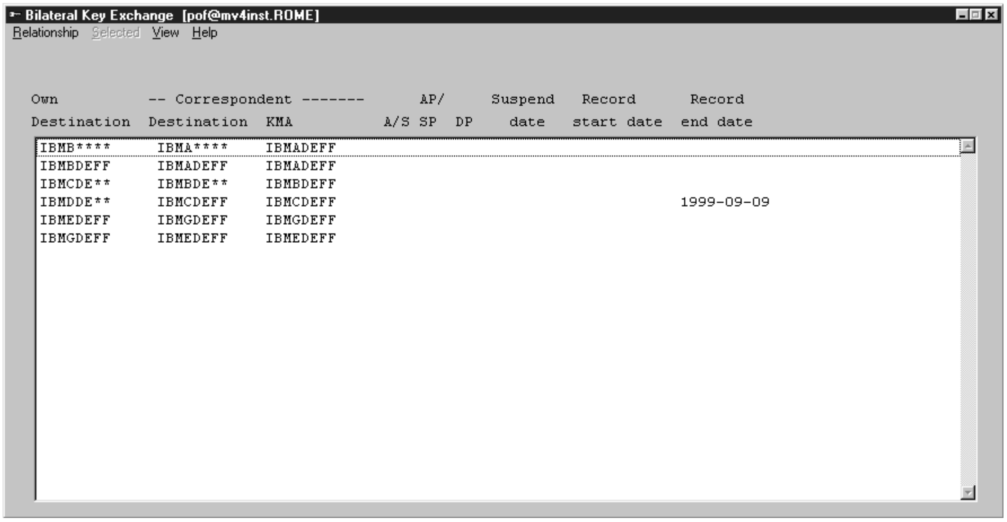
If you choose to continue to exchange paper-based keys, you can enter the key values manually.

To safeguard confidentiality, you should only use a bilateral key for a fixed period of time. Before a key expires, you exchange a replacement key with the correspondent.

The Bilateral Key Exchange Program should only be used concurrently by users that have different LTs assigned to in the MERVA User Maintenance program and do not have the **Bilateral Keys - All** right item assigned. Otherwise users might concurrently work with the same record, which can lead to confusion.

To begin the process of exchanging bilateral keys, invoke the **Bilateral Key Exchange** program from the **SWIFT USE UKMO** program group. The Bilateral Key Exchange window appears, as shown in Figure 28 on page 47.

The displayed list of correspondents depends on the user’s rights defined in User Maintenance. Refer to Appendix B for a detailed description.



```
Own          -- Correspondent -------       AP/     Suspend   Record      Record
Destination  Destination  KMA          A/S SP  DP    date   start date  end date
IBMB****     IBMA****     IBMADEFF
IBMBDEFF     IBMADEFF     IBMADEFF
IBMCDE**     IBMBDE**     IBMBDEFF
IBMDDE**     IBMCDEFF     IBMCDEFF                                      1999-09-09
IBMEDEFF     IBMGDEFF     IBMGDEFF
IBMGDEFF     IBMEDEFF     IBMEDEFF
```

Figure 28. The Bilateral Key Exchange Window

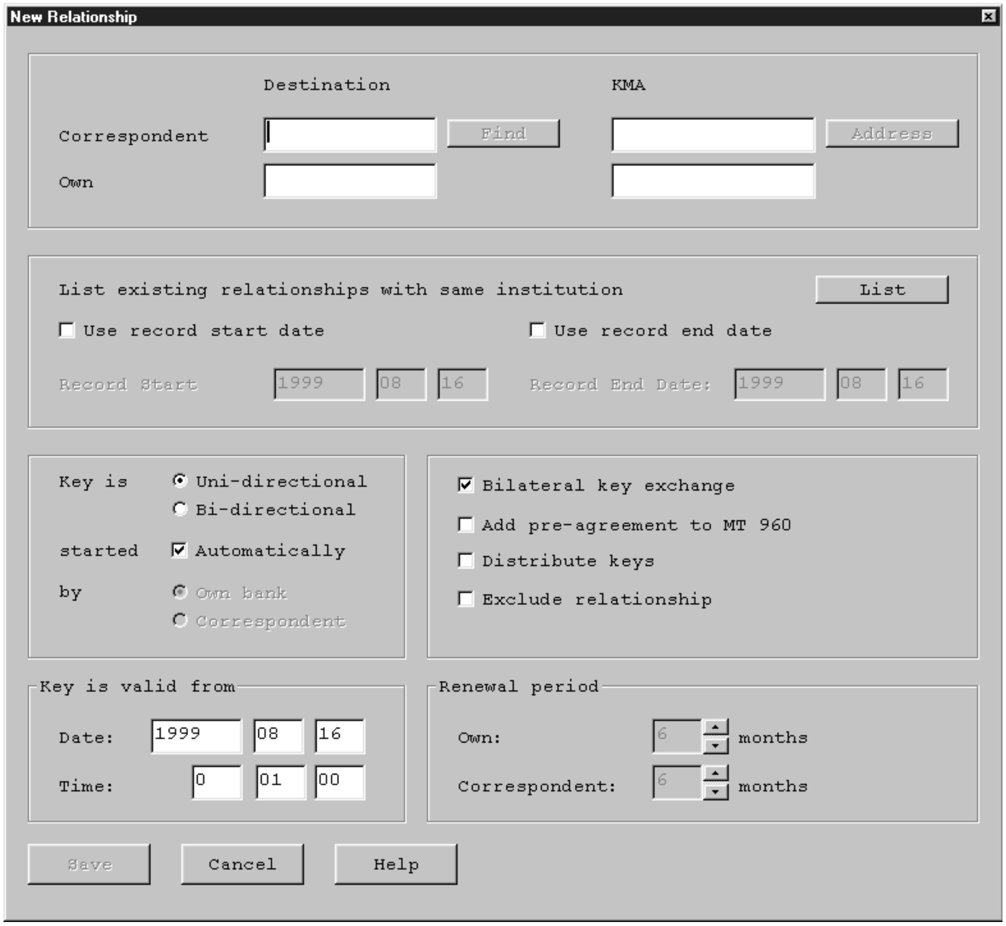
Creating a Pre-Agreement

A pre-agreement governs the terms under which the bilateral keys are exchanged. New and updated pre-agreements must be approved by a second UKMO.

Before exchanging bilateral keys with a correspondent for the first time, you must record the details of the correspondent in your database. If you are not sure whether you already have a record for the correspondent, try to find the correspondent in your database using the **Include** option. See “Managing the List of Correspondents” on page 66 for details.

Contact the correspondent to confirm the address of the key management authority (KMA) handling key exchange for the correspondent and to agree the terms of the pre-agreement.

To create a new correspondent record, select **New relationship** from the **Relationships** pull-down menu. The New Relationship window appears, as shown in Figure 29 on page 48.



	Destination	KMA
Correspondent		
Own		

List existing relationships with same institution

□ Use record start date

□ Use record end date

Record Start 1999 08 16

Record End Date: 1999 08 16

Key is ● Uni-directional ○ Bi-directional

started ☑ Automatically

by ● Own bank ○ Correspondent

☑ Bilateral key exchange

□ Add pre-agreement to MT 960

□ Distribute keys

□ Exclude relationship

Key is valid from

Date: 1999 08 16

Time: 0 01 00

Renewal period

Own: 6 months

Correspondent: 6 months

Figure 29. The New Relationship Window

For unique authenticaton keys you specify your and the correspondent's destination with the full 8-character BIC code (BBBBCCLL).

If authenticaton keys are shared by a group of destinations, you enter the destination in the following form:

- World-wide shared keys with bank code (first four characters of the SWIFT BIC) followed by four asterisks (BBBB****)
- Country-wide shared keys with bank code and country code (first six characters of the SWIFT BIC) followed by two asterisks (BBBBCC**).

When defining correspondent relationships using shared keys, you can exclude a specific destination from sharing based on the following rules:

1. Handling of an exception to a shared key: User AAAABEBB has a pre-agreement with user BBBBCHZZ using shared keys for all BBBB destinations. BBBBUS33 performs its own key management. For this case, you must specify a separate pre-agreement for BBBBUS33.

```
I)                                   II)
     Destination    KMA                   Destination    KMA
Cor.  BBBB****    BBBBCHZZ           Cor.  BBBBUS33    BBBBUS33
Own   AAAABEBB    AAAABEBB           Own   AAAABEBB    AAAABEBB
```

2. Handling of an exclusion for a shared key: BBBBNL2A should be excluded from the correspondent relationship between AAAABEBB and BBBB****. For this case, you must specify a separate pre-agreement for BBBBNL2A without any authenticaton keys.

   ```
        Destination    KMA                   Destination    KMA
   Cor.  BBBB****    BBBBCHZZ          Cor.   BBBBNL2A    BBBBCHZZ
   Own   AAAABEBB    AAAABEBB          Own    AAAABEBB    AAAABEBB
                                       --> Bilateral Key Exchange unselected
                                           No authentication keys
   ```

   This additional pre-agreement avoids that AAAABEBB can sent messages to BBBBNL2A using the shared keys.

   Specify this pre-agreement for manual exchange. Approve the pre-agreement. Do not enter any keys.

   If you select *Distribute Keys* for this pre-agreement to update the BK file on another system, you have to manually initiate the distribution.

Refer to the *S.W.I.F.T. Security Features Technical* for further details.

Fill in the destinations of:

- The correspondent.

  You can click on the **Find** push button to search your correspondents database for an entry for this correspondent.

  You can also enter only the first four characters of the BIC (the bank code) to search the correspondents database for all entries with that bank code. For example, if you enter **IBMB** and click on **Find**, the BIC address **IBMBDEFF** is found first. You can use the push buttons **>>** and **<<** to scroll forward or backward through the other addresses beginning with **IBMB** until the one you want is displayed. You can then click on **OK** to select the correspondent.
- The correspondent's Key Management Authority (KMA).

  Click on the **Address** push button to display a pop-up window that enables you to enter the address of the KMA and a comment, such as the telephone number of your contact person there.

  The address must be either 8 or 11 characters long.
- Your financial institution.
- Your KMA. This address must be either 8 or 11 characters long. If you do **not** have the *all bilateral key* right assigned in the User Maintenance, your KMA is restricted to the first 8 characters of the LT that is assigned to you in User Maintenance. The branch code can be chosen freely. If you do not have the *all BKE* right defined in the User Maintenance, refer to Appendix B.
- The **Start date** is used if you migrate from shared to unique keys. Then, you must specify a start date for the new unique relationship. The unique relationship then is used for authentication as of the start date. However bilateral keys have to be exchanged before the start date keys is reached.

  Note: This date is the date in the Greenwich Mean Time (GMT) time zone.
- The **End date** is used if you migrate from unique to shared keys. Then, you must specify an end date for the existing unique relationship. As long as the unique relationship is valid, keys can be exchanged for the generic relationship. When the end date of the unique relationship is reached, it is no longer used for authentication. Then the generic relationship is used.

Note: This date is the date in the Greenwich Mean Time (GMT) time zone.

- You can click on the **List** push button to see a list of all existing relationships between the specified institutions and their start and end dates.

After having defined the relationship and agreed the details of the pre-agreement with your correspondent, enter the following details about the pre-agreement into the New Partner Relationship window:

- **Uni-directional** or **bi-directional keys**

  If you select uni-directional, two keys are exchanged; one to authenticate messages received (the receive key) and the other to authenticate messages sent (the send key). The send key of the sender is the receive key of the receiver, and vice versa.

  If you select bi-directional, one key is exchanged: the same key is used to authenticate messages you send and those you receive.

  When own KMA is equal to correspondent KMA, a relationship must be defined as bi-directional.

- **Started Automatically**

  Only **one** of the relationships must be defined as ’started automatically’ if the own destination and correspondent destination are different, *but* the KMAs are the same. A protocol error message (MT964) occurs if ’started automatically’ is defined for both records.

  Example:

  ```
  I)                                  II)
       Destination    KMA                   Destination    KMA
  Cor.  BBBB****    BBBBBBBB          Cor.   BBBBBBBB    BBBBBBBB
  Own   BBBBBBBB    BBBBBBBB          Own    BBBB****    BBBBBBBB
   --> Started automatically           --> Not started automatically
  ```

  If you decide on automatic key exchange, MERVA automatically begins a new key exchange whenever necessary.

  If you choose not to implement automatic key exchange, you must make a note of when key exchange is necessary and begin the key exchange yourself (see “Starting Key Exchange Manually” on page 62).

  If you do opt for automatic key exchange, exchanges can still be monitored on an individual basis. See “Monitoring Bilateral Keys” on page 55 for information about monitoring key exchanges.

  You can use the MERVA routing mechanism to interrupt the automatic BKE process and perform manual authorization. Only *one* of the relationships must be defined as ’started automatically’ if the own destination and correspondent destination are different, *but* the KMAs are the same. A protocol error message (MT964) occurs if ’started automatically’ is defined for both records.

  Example:

  ```
  I)                                  II)
       Destination    KMA                   Destination    KMA
  Cor.  BBBB****    BBBBBBBB          Cor.   BBBBBBBB    BBBBBBBB
  Own   BBBBBBBB    BBBBBBBB          Own    BBBB****    BBBBBBBB
   --> Started automatically           --> Not started automatically
  ```

- **Own bank** or **Correspondent**

  If you exchange bi-directional keys, you must specify who is to begin the exchange:

  - Own bank. You begin the exchange.
  - Correspondent. Your correspondent begins the exchange.

  If you exchange uni-directional keys, you both begin the exchange for your own send key. In this case, this option cannot be selected.

  You must define **Own bank** if the own KMA is equal to the correspondent KMA. For relationships where the KMAs are the same but the own destination is different from the correspondent destination, **Own bank** must be specified in *both* relationships.

  Example:

```
I)                                  II)
     Destination    KMA                  Destination    KMA
Cor.  BBBB****    BBBBBBBB          Cor.   BBBBBBBB    BBBBBBBB
Own   BBBBBBBB    BBBBBBBB          Own    BBBB****    BBBBBBBB
 --> Specify own bank                --> Specify own bank
```

- **Bilateral key exchange**

  You can either allow the system to generate the bilateral keys or continue to take values from paper tables. In this case, you can type in the key manually.
- **Add pre-agreement to MT960**

  You can optionally send a copy of the pre-agreement with the first message of a key exchange. Your correspondent can then check that the pre-agreement that you are using is as expected.

  Note: This option is switched off when the first MT960 has been sent.
- **Distribute keys**

  Your keys are copied to a queue for routing to another MERVA system, for example, MERVA ESA. See “Distributing Keys to Other Systems” on page 63 and “Appendix B. Automatic Distribution of BK Data via MT999” on page 97.
- **Exclude relationship** If you do not want to exchange messages with the correspondent, you can mark the relationship as exclusion. No keys is exchanged for this relationship and no messages from/to the correspondent can be authenticated.

  You can not update this option. If you later want to have message traffic with this correspondent, you have to delete the relationship and create it again.
- **Key is Valid From**

  The time and date from which the first key will be effective.

  Notes:

  1. SWIFT recommends that:
     - The effective date for any future key be set to the third Sunday of the specified month, at 00:01:00 GMT
     - Key exchange be completed one week before this date and time.
  2. This date and time are in the Greenwich Mean Time (GMT) time zone.
- **Renewal Period**

  The period after which this key should be replaced. SWIFT recommends a period of validity of 6 months. This key remains valid, however, even if no new key is exchanged after this period.

  Note: Depending on the direction of the key, the own or the correspondent renewal period entry field is enabled; if the key is uni-directional, both fields are enabled.

Click on the **Save** push button to save the new correspondent record in the database.

Handling of Exclusions to Key-Sharing Rules

When using worldwide keys you should be aware of exceptions and exclusions. For example, in the case where user AAAABEBB has a pre-agreement with user CCCCDEDD, who uses a worldwide key for all branches except CCCCGB2L.

This pre-agreement looks as follows:

```
I)
     Destination    KMA
Cor.  CCCC****    CCCCDEDD
Own   AAAABEBB    AAAABEBB
```

If AAAABEBB has an account relationship with CCCCGB2L, a separate pre-agreement should be exchanged with CCCCGB2L.

Example for a separate pre-agreement used as exception:

```
II)
     Destination    KMA
Cor.  CCCCGB2L    CCCCGB2L
Own   AAAABEBB    AAAABEBB
```

Alternatively, CCCCDEDD or AAAABEBB may want to exclude CCCCGB2L, because either of them may not want authentication keys to exist between AAAABEBB and CCCCGB2L.

The problem here is that AAAABEBB has a valid key to authenticate messages to any destination CCCC****, using the worldwide key, including CCCCGB2L, but CCCCGB2L does not have a key with AAAABEBB. This means that a message could be sent by AAAABEBB and fail authentication when received by CCCCGB2L.

Example for a pre-agreement to exclude a specific destination:

```
III)
     Destination    KMA
Cor.  CCCCGB2L    CCCCGB2L
Own   AAAABEBB    AAAABEBB
```

Important settings in this pre-agreement are:

- ’Exclude Relationship’ is ON
- ’Distribute Keys’ is ON

  All other flags in that pre-agreement to exclude a specific destination are not relevant here.

  This correspondent relationship does not have any keys.

  Note: Also the pre-agreement to exclude a specific destination must be approved.

  If the authentication of financial messages is done on another system (for example, on a MERVA ESA) and this pre-agreement should become effective there also.

The distribution of the pre-agreement to exclude a specific destination, however, must be initiated manually by selecting **Send to distribution** from the selected pull-down. If that pre-agreement is deleted, a delete request is automatically distributed if ’Distribute Keys’ is ON.

The authentication works as follows:

For Sending: MERVA refuses any sending of messages for CCCCGB2L because there are no Authentication keys available.

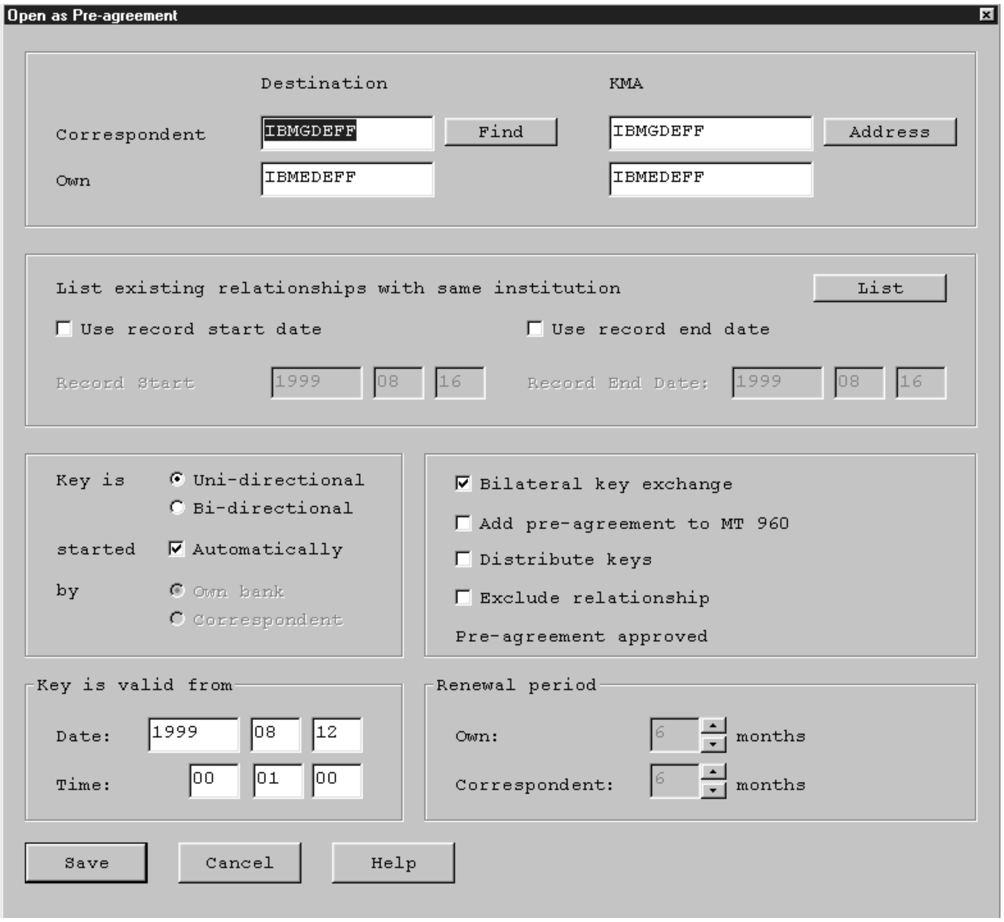
For Receiving: An entry is found but there is no Authentication key and an appropriate error message (no authentication key is found) appears.

Before key exchange can begin, the pre-agreement must be approved (see “Approving Pre-Agreements” on page 54).

Updating the Terms of an Existing Pre-Agreement

You can view and, if necessary, update the terms of the pre-agreement that you have made with a correspondent:

1. Select the correspondent from the list on the Bilateral Key Exchange window.
2. Select the **Open as pre-agreement** choice from the **Selected** pull-down menu. The Open as Pre-agreement window appears, as shown in Figure 30.



	Destination	KMA
Correspondent	IBMGDEFF	IBMGDEFF
Own	IBMEDEFF	IBMEDEFF

List existing relationships with same institution

□ Use record start date □ Use record end date

Record Start 1999 08 16 Record End Date: 1999 08 16

Key is ● Uni-directional
○ Bi-directional
started ☑ Automatically
by ● Own bank
○ Correspondent

☑ Bilateral key exchange
□ Add pre-agreement to MT 960
□ Distribute keys
□ Exclude relationship
Pre-agreement approved

Key is valid from
Date: 1999 08 12
Time: 00 01 00

Renewal period
Own: 6 months
Correspondent: 6 months

Figure 30. The Open as Pre-Agreement Window

3. Review and update the terms of the pre-agreement as necessary.
4. Click on the **Save** push button.

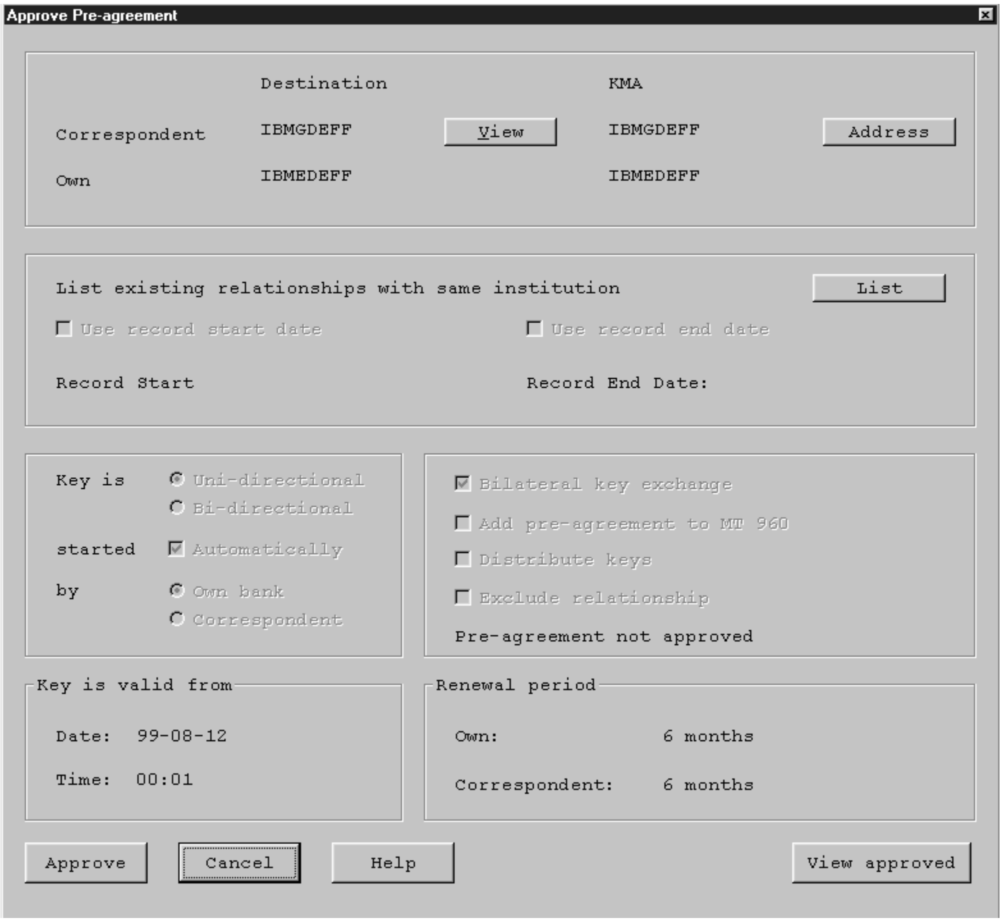
If there is already an approved pre-agreement, this remains valid until the terms of the updated pre-agreement are approved. You can see the terms of the previous pre-agreement by clicking on the **View approved** push button.

Approving Pre-Agreements

Before bilateral key exchange can begin, pre-agreements must be approved. This procedure should be performed by a second UKMO.

To approve a pre-agreement:

1. Select the correspondent from the list on the Bilateral Key Exchange window.
2. Select **Approve pre-agreement** from the **Selected** pull-down menu.
3. If the pre-agreement has not already been approved, the Approve BKE Pre-Agreement window shown in Figure 31 is displayed.



	Destination	KMA
Correspondent	IBMGDEFF	IBMGDEFF
Own	IBMEDEFF	IBMEDEFF

List existing relationships with same institution

□ Use record start date   □ Use record end date

Record Start   Record End Date:

Key is ● Uni-directional ○ Bi-directional

started ☑ Automatically

by ● Own bank ○ Correspondent

☑ Bilateral key exchange
□ Add pre-agreement to MT 960
□ Distribute keys
□ Exclude relationship

Pre-agreement not approved

Key is valid from

Date: 99-08-12

Time: 00:01

Renewal period

Own: 6 months

Correspondent: 6 months

Figure 31. The Approve Pre-Agreement Window

4. When all the values are correct, click on the **Approve** push button.

To approve more than one pre-agreement, select the correspondents from the list on the Bilateral Key Exchange window, then select **Approve pre-agreement** from the **Selected** pull-down menu. A window containing all selected records for approval is displayed. Click on the **Approve** push button to approve the pre-agreements. The **Status** column contains the following:

Already approved
The pre-agreement is already approved.

Record approved
The pre-agreement has been successfully approved.

Error approving record
An error occured while approving the record.

Note: The pre-agreement information is not be displayed during the approval process. Ensure that the pre-agreements terms of the selected records are correct.

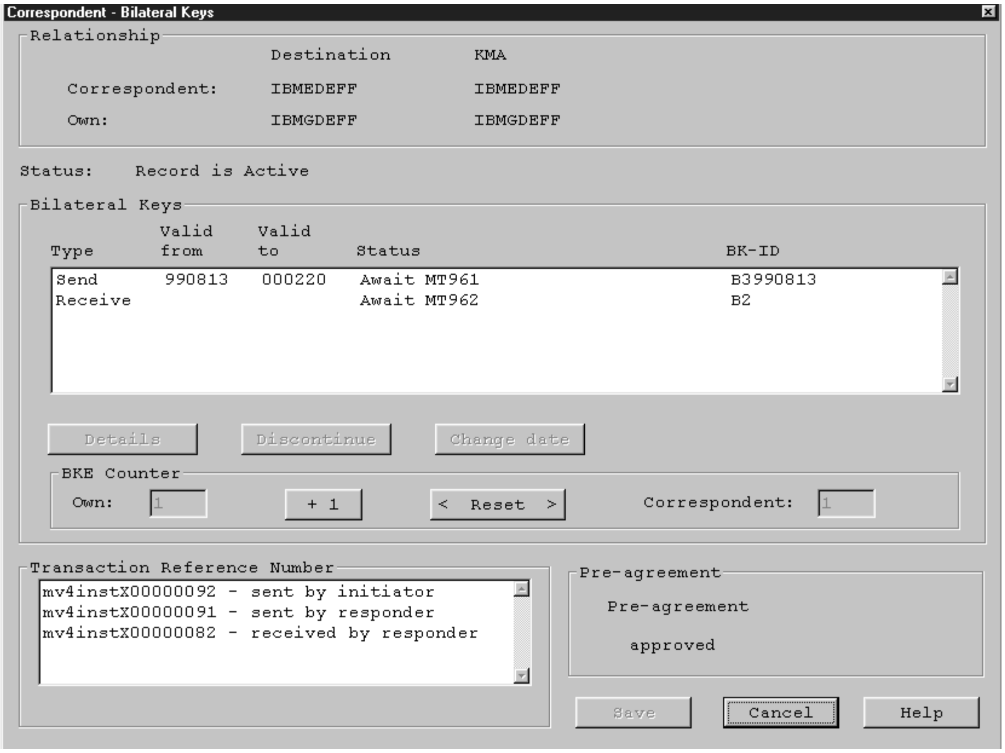
Monitoring Bilateral Keys

For each correspondent, MERVA provides you with up-to-date information about the keys you share with the correspondent, the status of each key exchange, and details about each key. You can also print out this information (see "Printing Correspondents' Information" on page 69).

To view a summary of bilateral key information for a correspondent:

1. Select the correspondent whose keys you want to inspect from the list on the Bilateral Key Exchange window.

   If you cannot find the entry for the correspondent, it may be because only certain types of correspondent are included in the list. Ensure you have included the correspondent in the list (see "Managing the List of Correspondents" on page 66).
2. Select **Open as bilateral keys** from the **Selected** pull-down menu. The Correspondent - Bilateral Keys window appears (see Figure 32 on page 56).



```
Correspondent - Bilateral Keys
Relationship
                     Destination       KMA
   Correspondent:    IBMEDEFF          IBMEDEFF
   Own:              IBMGDEFF          IBMGDEFF

Status:   Record is Active

Bilateral Keys
              Valid     Valid
  Type        from      to        Status                           BK-ID
  Send        990813    000220    Await MT961                      B3990813
  Receive                         Await MT962                      B2

BKE Counter
  Own: 1                                          Correspondent: 1

Transaction Reference Number                  Pre-agreement
mv4instX00000092 - sent by initiator            Pre-agreement
mv4instX00000091 - sent by responder              approved
mv4instX00000082 - received by responder
```

Figure 32. The Correspondent - Bilateral Keys Window

The Relation area of the window shows you and your correspondent's destination and KMA addresses.

The Bilateral Keys area of the window shows you:

- The type of keys you share with the correspondent:
  - **Send** or **Receive** if the pre-agreement specifies uni-directional keys
  - **Snd/Rcv** if bi-directional keys are in use
- The 'valid to date'. There is no expiration date associated with bilateral keys; this 'valid to date' is only a reminder to exchange new bilateral keys before this date. The current key is used to authenticate incoming and outgoing messages until a new bilateral key is exchanged or manually entered and becomes current.

  Note: This date is the date in the Greenwich Mean Time (GMT) time zone.
- The current status of each key (see "Bilateral Key Status" on page 57).
- The value of the 16-character BK identifier (BK-ID).The BK-ID has the following format:
  - The first character is either B (Bilateral) or M (Manual)
  - The second character is the BK type, as defined by SWIFT
  - Characters 3 to 8 inclusive are the date
  - Characters 9 to 16 inclusive are the key check value.

    If you encounter problems communicating with your correspondent, check the key check value, which should be identical to your correspondent's.

Note: When you perform any activity on this window, the database remains unchanged until you click on the **Save** push button. Use the **Cancel** push button to leave the window without any changes.

Bilateral Key Status

From the time you first propose the exchange of a new bilateral key with a correspondent until the bilateral key being replaced, each bilateral key passes through a number of states. The current status of a bilateral key is displayed in the Status column of the Correspondent - Bilateral Keys window.

Because the bilateral key status is based on Greenwich Mean Time (GMT), it is essential that the correct time-zone be specified for your Windows NT system.

The possible status values are as follows:

Status **Meaning**

Await MT961 The MT960 (BKE Initiation Request) message has been sent to the correspondent. You are waiting for the exchange to be accepted.

Timeout MT961
The MT961 (the BKE Initiation Acknowledgement) message has not arrived within the permitted period, for example, 7 days (the default).

The period after which timeout occurs can be customized. See the *MERVA USE & Branch for Windows NT Installation and Customization Guide*.

Await MT962 You are responding to an exchange request initiated by a correspondent. The MT961 (BKE Initiation Acknowledgement) message has been sent to the correspondent. You are waiting for the new key of the correspondent to arrive.

Timeout MT962
You are responding to an exchange request initiated by a correspondent. The MT962 (BKE Key Service Message) has not arrived within the permitted period, for example, 7 days (the default).

The period after which timeout occurs can be customized. See the *MERVA USE & Branch for Windows NT Installation and Customization Guide*.

Await MT963 The MT962 (BKE Key Service Message) has been sent to the correspondent. You are waiting for an acknowledgement that the key has been received safely.

Timeout MT963
The MT963 (BKE Key Acknowledgement) message has not arrived within the permitted period, for example, 7 days (the default).

The period after which timeout occurs can be customized. See the *MERVA USE & Branch for Windows NT Installation and Customization Guide*.

Await MT967 The MT966 (BKE Discontinuation Request) message has been sent to the correspondent. You are waiting for an acknowledgement that the key or keys are to be discontinued.

Timeout MT967
The MT967 (BKE Discontinuation Acknowledgement) message has not arrived within the permitted period. The key can no longer be used for sending messages. When receiving messages, the user is informed that a discontinued key was used. See page 60 for details.

The period after which timeout occurs can be customized. See the *MERVA USE & Branch for Windows NT Installation and Customization Guide*.

Future
The new key has been safely received, and will become current on the date shown (or sooner in an emergency).

Current
The key is currently being used to authenticate financial messages.

Note: The date shown is the date at this time in the Greenwich Mean Time (GMT) time zone; the date in your time zone might differ.

Previous
The key has been replaced by another current key.

Discontinued
The key can no longer be used for sending messages. When receiving messages, the user is informed that a discontinued key was used. See page 60 for details.

Discontinue Error
A protocol error occurred during discontinuation. Display the Diagnosis Log for more information. See page 60 for details.

Error detected by CBT
An error has been detected by MERVA that prevents message exchange continuing.

Error detected by correspondent
An error has been detected by your correspondent that prevents message exchange continuing and an appropriate MT964 (BKE Protocol Error) or MT965 (BKE Key Error) message has beenreceived from the correspondent informing you of this.

Figure 33 on page 59 illustrates the various states in the life cycle of a bilateral key.

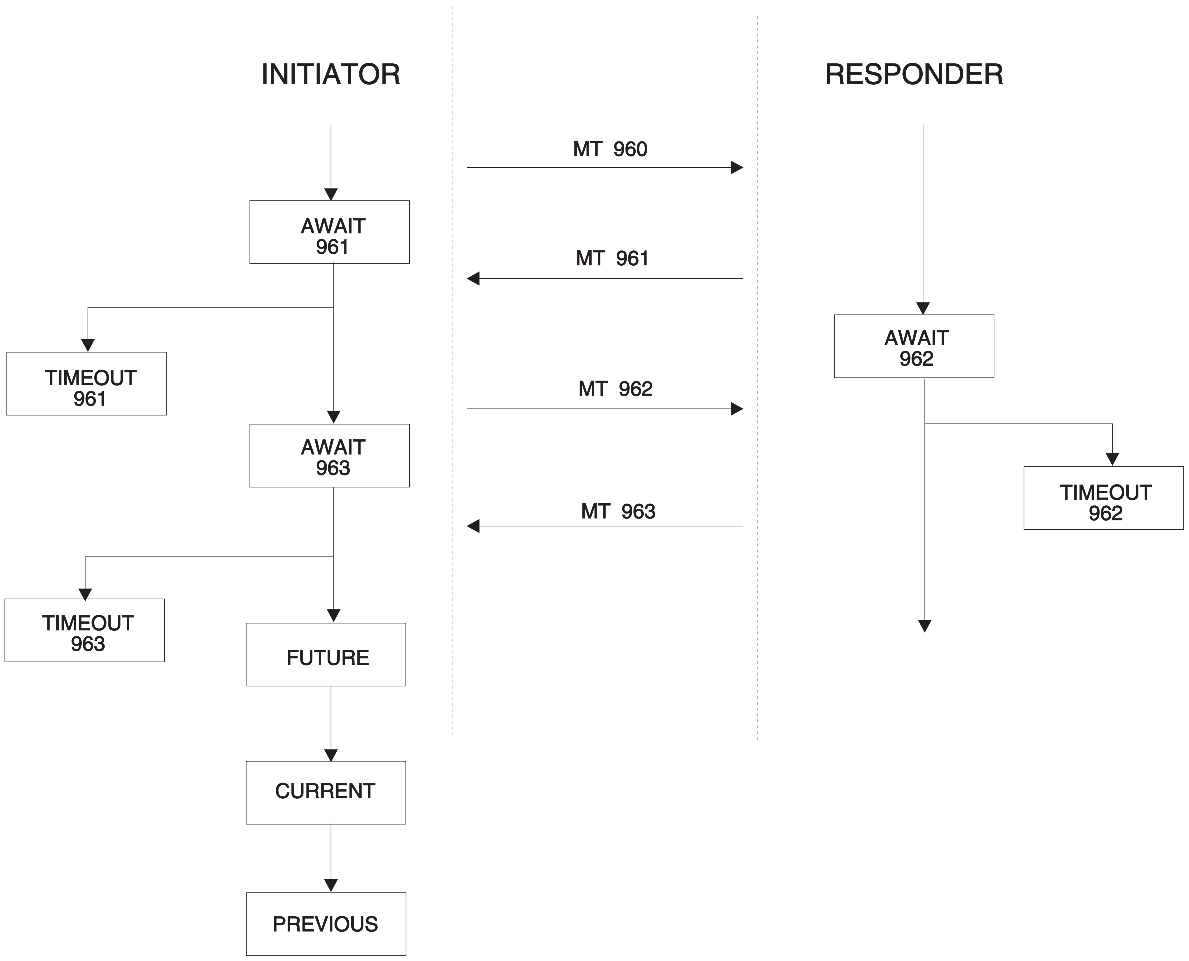


Figure 33. The Life Cycle of a Bilateral Key

A TIMEOUT status shows you that a problem has occurred during the key exchange process. Make a note of the transaction reference number (TRN) for the key (see “Dealing with Transaction Reference Numbers” on page 62), then contact your correspondent to find out what the problem is. Once you have solved the problem, you must initiate a new key exchange manually. See “Starting Key Exchange Manually” on page 62 for details of how to do this.

An ERROR status indicates that an error has occurred. These messages are not dealt with automatically; you must correct the error yourself, for example, using the Incoming MT960/MT966 program (see “Dealing with Incoming MT960 and MT966s” on page 70).

Note that if your own destination and the correspondent destination are identical, the BKE protocol is shortened and, with the exception of MT960 messages, no MT96x message is sent or received via the SWIFT Network. Instead, the bilateral key is generated immediately in the card reader and stored in the database.

Entering and Displaying Manual Keys

If the pre-agreement specifies that manual keys are to be used to authenticate financial messages, that is, using the current paper table technology, you can enter or change the bilateral keys manually. In this case:

- The **Bilateral key exchange** option is not selected on the New Partner Relationship or Open As Pre-Agreement window.

- A **New** push button appears instead of the **Details** push button on the Correspondent - Bilateral Keys window.

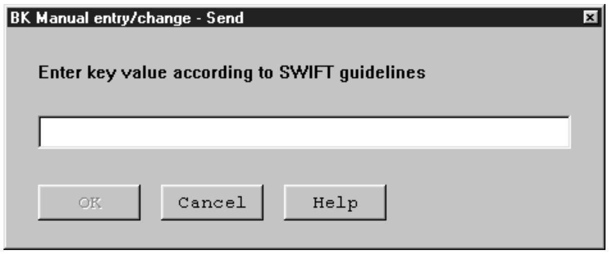
To display an existing bilateral key:

1. Select a key from the Bilateral Keys area of the Correspondent - Bilateral Keys window.
2. Click on the **Details** push button. The BK - Manual Entry/Change window is displayed, as shown in Figure 34.
3. Click on the **OK** push button to return to the Correspondent - Bilateral Keys window.

To enter the bilateral key manually:

1. Click on the **New** push button on the Correspondent - Bilateral Keys window.

   The BK - Manual Entry/Change - Send window is displayed, as shown in Figure 34.



BK Manual entry/change - Send

Enter key value according to SWIFT guidelines

Figure 34. The BK - Manual entry/change - Send Window

2. Type in the value of the key. The keys entered must either be 16 characters long, using all the digits 0 to 9 **and** the letters A to F in any order, or exactly 32 characters long, using any of the digits 0 to 9 **or** the letters A to F. For 32 character long keys, also the following recommendations apply:
   - Each character should only appear once in the first half of the key.
   - Each character should only appear once in the second half of the key.
   - The first half of the key should be different from the second half.
3. Click on the **OK** push button to return to the Correspondent - Bilateral Keys window.

Click on the **Save** push button to store any changes or new keys in the database.

Discontinuing a Key in an Emergency

Whenever you suspect that the confidentiality of a bilateral key has been compromised, you must immediately discontinue using the key.

In an emergency, you can discontinue a key to immediately stop sending messages to this correspondent using that key. If you want to continue message exchange, change the date of validity of another key you share with the correspondent.

To discontinue a key:

1. Select the correspondent whose key you want to discontinue on the Bilateral Key Exchange window.
2. Select **Open as bilateral keys** from the **Selected** pull-down menu. The Correspondent - Bilateral Keys window is displayed, showing you the keys you share with the correspondent and the current status of the keys.
3. Select the key you want to discontinue.

4. Click on the **Discontinue** push button.
5. Select **Yes** on the message window that is displayed to confirm your action. The status of the selected key changes to DISCONTINUED.
6. Click on the **Save** push button to update the database and generate the Discontinue message.

If a future key is available, you may change the validity date for the future key, so that financial transactions can continue to be exchanged.

Note: If you have discontinued all your keys with a correspondent because you do not want to allow financial transactions at present, you should check and, if necessary, modify the pre-agreement so that key exchange does not take place automatically.

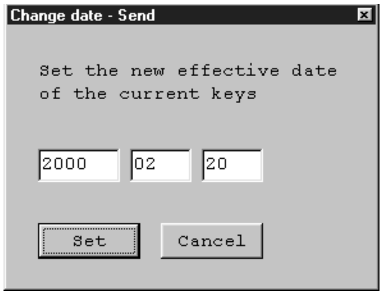
Changing a Future Key’s Effective Date

If you discontinue a current key, but you and the correspondent want to continue sending and receiving financial transactions, then you can immediately switch to using a future key. You bring the date of validity of the future key forward to the current date, or whatever date you agree with the correspondent. At the specified time the status of the key changes from FUTURE to CURRENT, and you can use the key.

Note: The date entered for this key is the date in the Greenwich Mean Time (GMT) time zone.

To substitute a future key for the discontinued key:

1. On the Correspondent - Bilateral Keys window, select the keys with the status FUTURE.
2. Click on the **Change date** push button. The Change Date window appears, showing the date of validity for the key.



Set the new effective date of the current keys

2000 02 20

Figure 35. The Change Date - Send Window

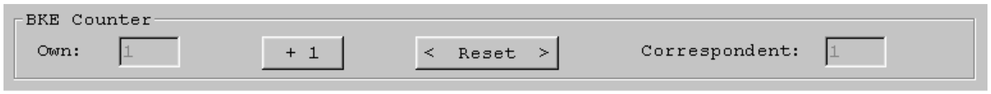
3. Change the new effective date to whatever you have agreed with your correspondent. From this time on, the key will be current.
4. Click on the **Set** push button to change the effective date.
5. Click on the **Save** push button to update the database.

Having changed the status of your future key to current, you now have no key available to substitute in an emergency. You do not have to wait for an automatic BKE start, you can start the key exchange manually.

Managing Key Exchange Counters

Counters are used to confirm that you and your correspondent receive the messages required for a key exchange. The key exchange counters are a safeguard to ensure that key exchanges are not lost and that no illegal exchanges take place.

The value of the counter tells you how many keys you have exchanged with each other. The counter is incremented each time you exchange a key. The values for the counter, as stored by your correspondent and as stored by you, are shown in the BKE counters area of the Correspondent - Bilateral Keys window:



BKE Counter

Own: 1

Correspondent: 1

Figure 36. The BKE Counters Area of the Correspondent - Bilateral Keys Window

A mismatch of the BKE counters is detected only at the responding side of an MT962 message. This causes an MT965 (Error: ERF /P) to be sent to the initiator of the exchange.

Inform your correspondent that you both need to reset the counters and begin a new key exchange. To do this, click on the **Reset** push button.

The value displayed in the **Own BKE Counter** field is reset to **1**. Click on the **Save** push button to update the database.

Dealing with Transaction Reference Numbers

When there are problems with a key exchange and you need to contact the correspondent, you can use the transaction reference number (TRN) to identify the BKE message causing problems. The TRNs for the most recent messages you have sent and received are displayed in the **Transaction reference numbers** list on the Correspondent - Bilateral Keys window.

Your correspondent stores the identical numbers, so you can quickly confirm you are both dealing with the same message.

The TRN of a BKE message is identical to the message reference number (MRN) of a message to which the BKE message refers. The MRN is assigned by MERVA. You can use the contents of this TRN to retrieve a message directly. See the *MERVA USE & Branch for Windows NT User's Guide* for an explanation of the MRN and a description of how to use these functions.

Starting Key Exchange Manually

If specified in the pre-agreement, MERVA starts key exchange for you automatically. If, however, you do not use automatic exchange, or you need to exchange an extra key not covered by the pre-agreement, you can start an exchange manually.

If you decide to stop exchanging keys automatically with an existing correspondent, you should inform the correspondent and modify the pre-agreement accordingly.

To start an exchange manually:

1. Select one or more correspondents from the list on the Bilateral Key Exchange window.

2. From the **Selected** pull-down menu, select **Start BKE**.

The window shown in Figure 37 is displayed.

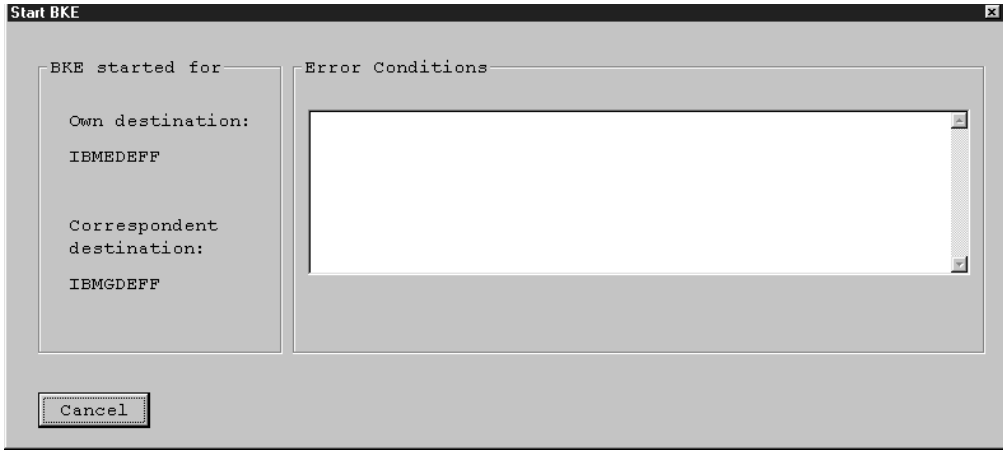


Figure 37. Starting Bilateral Key Exchange Manually

A key exchange is started with the selected correspondent. If you have selected more than one correspondent, you can click on the **Stop** push button at any time to stop the start of BKE for more correspondents. Any errors that occur during the exchange are displayed in the **Error Conditions** list box. These could include, for example:

- Pre-agreement not approved
- This system is not the initiator of the exchange
- No certificate is available.

Distributing Keys to Other Systems

You can copy your keys to a message queue to allow the keys to be routed to another system, for example using MERVA Link.

Routing to other systems is performed automatically if the option **Distribute Keys** on the BKE Pre-agreement window is selected. See “Creating a Pre-Agreement” on page 47 for details. You can distribute keys to other MERVA systems.

To send keys to another system:

1. Select one or more correspondents whose keys you want to send, by clicking on the entry for the correspondent on the Bilateral Key Exchange window.
2. Select **Send to distribution** from the **Selected** pull-down menu.

   The keys are routed to the MERVA queue that was defined during installation or customization.

Note: Only keys from correspondent relationships with approved pre-agreements can be distributed.

Deleting a Correspondent

If you do not intend to exchange financial messages with a correspondent in the future, you can delete the correspondent relationship from the database.

To delete one or more relationships from your database:

1. From the Bilateral Key Exchange window, select the relationship or relationships you wish to delete. The selected correspondent relationships are highlighted.
2. Select **Delete** from the **Delete** submenu of the **Selected** pull-down menu.

   A confirmation window appears for the correspondent relationship(s), asking you whether you really want to delete the relationship(s). If you confirm the deletion, the highlighted relationship is marked as ′delete pending′ and a ′DP′ appears on the Bilateral Key Exchange Window.

   Nevertheless this relationship is used for authentication and treated like any other correspondent relationship.
3. Select **Approve** from the **Delete** submenu of the **Selected** pull-down menu to delete the correspondent relationship from the database, or
4. Select **Undelete** from the **Delete** submenu of the **Selected** pull-down menu to reset the ′delete pending′ status of the relationship.

To delete all correspondent relationships from the database, select **Delete all** from the **Delete All** submenu of the **Relationships** pull-down menu. The **Undelete All** and **Approve Delete All** functions on the same submenu work similar.

Notes:

1. To approve a pending delete of a relationship with **Approve** or **Approve Delete All** you must have the **Bilateral Keys - Approve pre-agreement** right assigned to you in the **MERVA User Maintenance** program.
2. To undelete a relationship, you need the right **Bilateral Keys - Maintain**, or the right **Bilateral Keys - Approve pre-agreement**.
3. If you do not have the **Bilateral Keys - All** right, Delete All, Undelete All, and Approve Delete All change only those correspondent relationships that belong to the LT assigned to you in the **MERVA User Maintenance** program.

If you do not have the **Bilateral Keys - All** right defined in the **MERVA User Maintenance** program, refer to “Appendix C. Working with Several KMAs” on page 99.

If a correspondent relationship is deleted by the user, it is marked as deleted, but still kept in the database. If the BK keys are distributed to another branch as partial file, then the information about the deleted relationships is also added to the distribution file. After a customizable period of time the USE batch process physically deletes the relationships that are marked as deleted from the database automatically. The period of time the deleted relationships are held in the database is by default 30 days.

This default can be changed by the environment variable **ENM_KEEP_DEL**.

If you do not have to distribute deleted relationships to other branches, you can delete BKE records immediately by setting the variable **ENM_KEEP_DEL=0**. After **Delete Approve** or **Delete All Approve**, the BKE program removes the records from the database.

Set this variable as a system variable in your system environment. To change the value of the variable:

1. Log on as system administrator.
2. Click **Start** → **Settings** → **Control Panel** → **System** → **Environment**.
3. Place your cursor in **System Variables**.
4. In the field **Variable**, type **ENM_KEEP_DEL**.

5. In the field **Value**, type **nnn** where **nnn** is a figure of up to three digits. This value specifies how long the deleted relationships are kept in the database.

   nnn=60, for example, specifies that a deleted relationship is kept in the database for 60 days. Then, it is removed from the database.

   nnn=0, for example, specifies that BKE program removes the relationships that are immediately triggered by the functions **Delete Approve** or **Delete All Approve**. You cannot distribute deleted relationships via the partial distribution file **SWIFT Authentication Key Distribution**.

After you change the variable, you have to:

1. Restart your computer to make the changes effective.
2. Restart your MERVA system.

Suspension and Re-Activation of Relationships

If there is any reason for temporary disabling the financial message traffic with a correspondent, like a disaster, a war, a blockage, an embargo or business suspension, the record can be suspended.

A suspended record is not used for authentication, but new keys can be exchanged for the record as usual.

There are two types of suspension:

1. **Immediate Suspension**

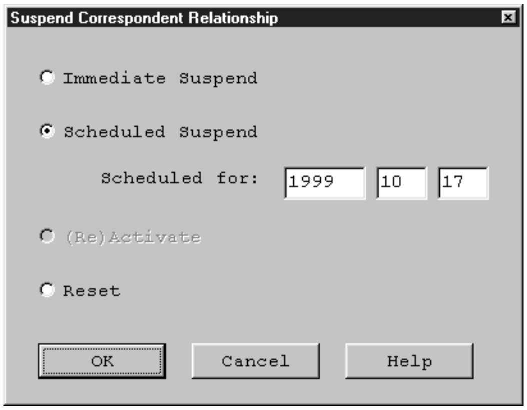
   Immediate suspension makes sense if there is an unexpected critical situation, and you want to temporarily disable financial message traffic immediately. If the relationship is suspended, you can think about the situation and decide how to further go on with this correspondent.
2. **Scheduled Suspension**

   Scheduled suspension makes sense if there is a planned stop of financial message traffic.

To suspend a correspondent relationship:

1. From the Bilateral Key Exchange window, select the relationship you wish to suspend. The selected correspondent relationship is highlighted.
2. Select **Suspend/Re-activate** from the **Suspend** submenu of the **Selected** pull-down menu.

   The Suspend Correspondent Relationship Window appears as shown in Figure 38 on page 66.



Suspend Correspondent Relationship

○ Immediate Suspend

● Scheduled Suspend

Scheduled for: 1999 10 17

○ (Re)Activate

○ Reset

Figure 38. The Suspend Correspondent Relationship Window

3. Decide whether to suspend the relationship immediately or scheduled. If the relationship is to be suspended at a future date, enter the date into the date selection box.

   Note: This date is the date in the Greenwich Mean Time (GMT) time zone.

   If the suspension of a relationship is to be cleared, the relationship has to be re-activated.
4. Press OK. The state of the relationship changes to ’Suspend pending’ or ’Re-activate pending’.

Suspension and re-activation both have to be approved in a separate step. Therefore:

1. From the Bilateral Key Exchange window, select the relationship where a Suspend or Re-activate is pending. If a suspend or reactivate is pending, an ’SP’ or ’AP’ appears in the column ’SP / AP’ of the Bilateral Key Exchange window.
2. The selected correspondent relationship is highlighted.
3. Select **Approve** from the **Suspend** submenu of the **Selected** pull-down menu.

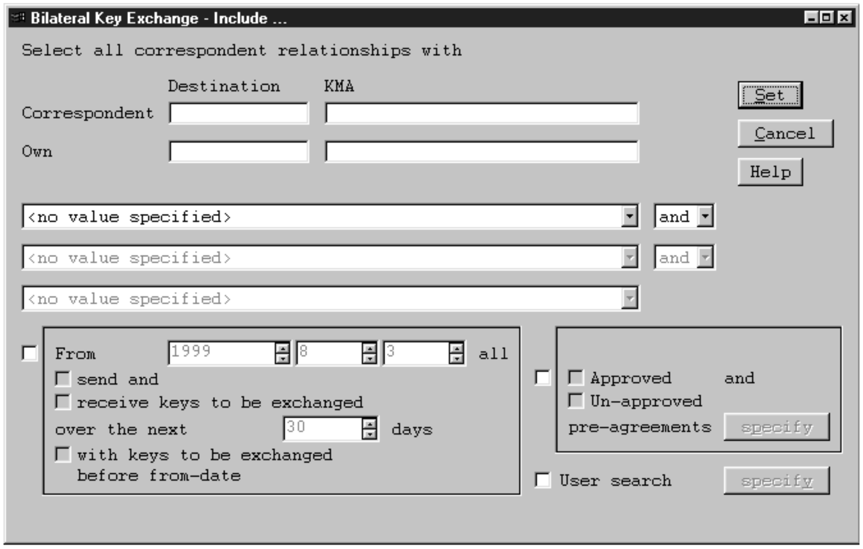
   The Suspend Correspondent Relationship window appears as shown in Figure 38.
4. If the state shown in the window is correct, press the **Approve** push button, otherwise press Cancel to leave the window.
5. The status of the relationship changes in the Bilateral Key Exchange window, if you approved the pending Suspend/Re-Activate.

Managing the List of Correspondents

You can use the choices on the **View** pull-down menu to change the appearance of the list of correspondents.

The correspondents in the list are taken from your BKE database. If you have hundreds of correspondents, it is not practical to display all the correspondents at the same time. Therefore, you can define that only certain correspondents are included in the list.

Select **Include** from the **View** pull-down menu.



Bilateral Key Exchange - Include ...

Select all correspondent relationships with

Destination KMA

Correspondent

Own

<no value specified> and

<no value specified> and

<no value specified>

□ From 1999 8 3 all

□ send and

□ receive keys to be exchanged

over the next 30 days

□ with keys to be exchanged before from-date

□ □ Approved and

□ Un-approved

pre-agreements

□ User search

Figure 39. The Bilateral Key Exchange - Include Window

The following filter criteria are available:

- You can filter the list of correspondents to display only relationships where the BICs match the specified filter:

  In the **Destination** and **KMA** fields of **correspondent** and **own** enter the BICs to be included into the list of correspondents.

  You can use substitution characters in the BICs to obtain a best match.

  – The percent character (%) is used to represent any sequence of zero or more characters.

    For example, you can include all BICs for which the bank name starts with **FED** by entering **FED%**.

  – The underscore character (_) can be used to represent a single character.

    For example, you can include all BICs for country **CC** by entering **____CC%**.

- You can also reduce the list by selecting items from the criteria list boxes. For example, select **Timeouts** to display only the correspondent relationships for which any key has a Timeout status.

  These criteria can be connected logically with AND or OR.

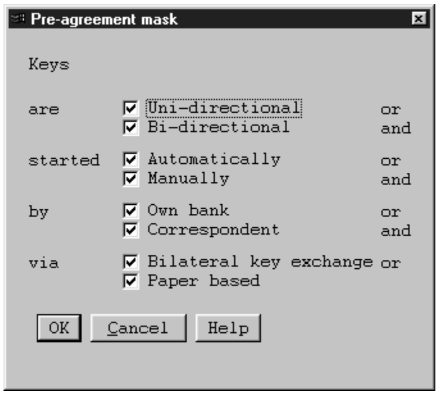
  Note: Only the criteria up to the first blank criterion are used.

- You can also display a list of the correspondent relationships where a new **send** or **receive** key is to be exchanged. For example, to find all relationships where keys have to be exchanged in the next month, do the following:

  1. Mark the check box in front of the group box that contains the query, to enable the group box.
  2. Set the date to the 1st of the next month.
  3. Select both, the **Send** and **Receive** check box.
  4. Set the number of days to search to the number of days of the next month.
  5. Do not check the last check box, that says ’with keys to be exchanged before from-date’. This check box should only be checked to see also keys that should have been exchanged in the past (before the specified from-date), but where no new key exists.

- You can specify to include only relationships with specific pre-agreement terms. Therefore:

1. Mark the check box in front of the pre-agreement group box. The group box is enabled.
2. Select whether the pre-agreement terms should appear only in the approved or unapproved pre-agreement, or in both pre-agreements. The **specify** pre-agreement push button gets enabled.
3. Push the specify pre-agreement push button. The Pre-agreement Mask Window shown in Figure 40 is displayed.



Keys

are ☑ Uni-directional or
☑ Bi-directional and

started ☑ Automatically or
☑ Manually and

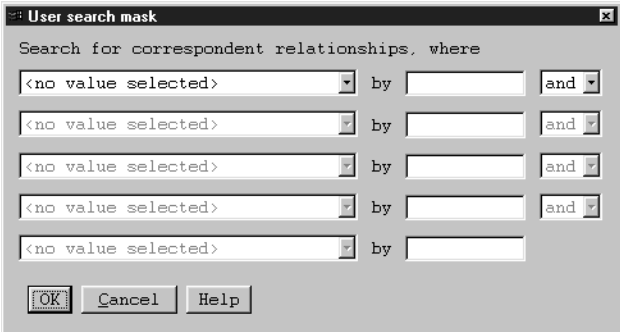
by ☑ Own bank or
☑ Correspondent and

via ☑ Bilateral key exchange or
☑ Paper based

Figure 40. The Pre-agreement Mask Window

Select the terms and press OK to save the changes or Cancel to leave the window without changing the pre-agreement mask.

- You can also only include relationships into the list of correspondents, where specific users have done actions like updating the pre-agreement or approving the pre-agreement. Therefore:
  1. Mark the User Search check box. The **specify** push button gets enabled.
  2. Push the specify push button. The User Search Mask window shown in Figure 41 is displayed.



Search for correspondent relationships, where

<no value selected> by and
<no value selected> by and
<no value selected> by and
<no value selected> by and
<no value selected> by

Figure 41. The User Search Window

  3. Select the user action from the first list box and enter the User ID in the entry field. The next row gets enabled. You can now specify the next condition, if you want to.

     Note: Only the rows up to the first blank row are used.
  4. Press OK to save the changes or Cancel to leave the window without changing the user search mask.

Click on the **Set** push button to update the list of correspondents on the Bilateral Key Exchange window.

Note: The values on the Include window are saved, so that the same list of correspondents is displayed when you start the Bilateral Key Exchange program the next time.

The following additional choices are available on the **View** pull-down menu:

Settings
You can use the Settings choice to set the following options:

- Whether information messages are displayed.
- Whether a warning is issued if the correspondent record you are working with is not currently stored in the correspondent database.
- Whether the name of the correspondent should be included in the primary display panel. This setting is activated upon the next start of the program.

  Note: If many correspondent records exist, including correspondent name information slows down performance. Therefore you should only use this setting if really needed.
- Whether the list of correspondents is filled when starting the Bilateral Key Exchange program or not. If you do not want to wait for the list of correspondents to be filled with result of the last include criteria, but immediately specify your own include criteria, do not fill the list of correspondents at startup.
- How many relationships are read from the database and filled into the list of correspondents during refresh, or when reaching the end of the displayed list.

Refresh
Updates the data in the list.

Printing Correspondents' Information

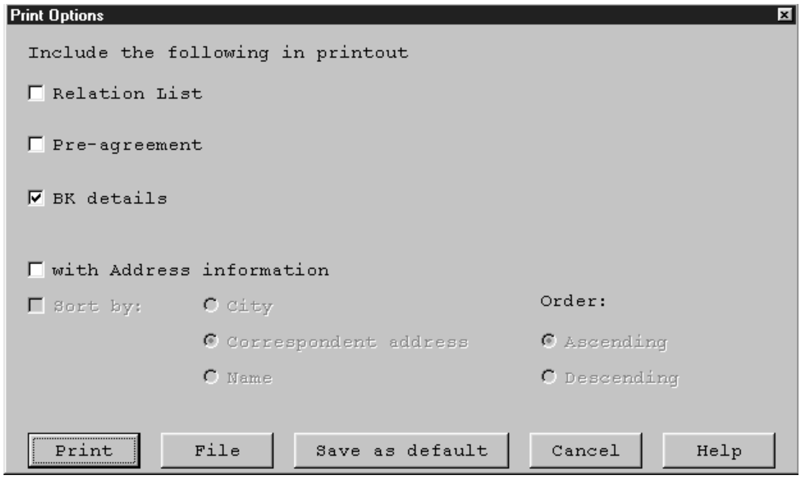
You can send information about your correspondents to a printer. You can choose to print a list of your correspondents, details of those correspondents currently selected in the Bilateral Key Exchange window, or the details of a single correspondent only:

- To print the list of all correspondents currently stored in the database, select **Print all** from the **Relationships** pull-down menu.

  If the **Bilateral Keys - All** right item is not assigned to you, **Print All** allows you to print all the correspondent relationships that belong to the LT assigned to you in the **MERVA User Maintenance** program.

  If you do not have the **Bilateral Keys - All** right defined in the **MERVA User Maintenance** program, refer to “Appendix C. Working with Several KMAs” on page 99.
- To print all the data held in the database for the selected correspondents (the list of keys, the details for each key, and the pre-agreement), first select the correspondents you require. Then select **Print** from the **Selected** pull-down menu.

In both cases, the Print window appears, as shown in Figure 42 on page 70.



Print Options

Include the following in printout

□ Relation List

□ Pre-agreement

☑ BK details

□ with Address information

□ Sort by: ○ City ● Correspondent address ○ Name

Order: ● Ascending ○ Descending

Figure 42. The Bilateral Key Exchange - Print Window

Decide whether you want to print the data immediately or save it to a file:

- To print the data immediately, click on the **Print** push button.

  Note: Ensure that a default printer is assigned to you. For a description on how to assign a default printer, refer to the *MERVA USE & Branch for Windows NT User's Guide*.
- To save the data to a file, click on the **File** push button. A window appears allowing you to select an existing file or enter a file name to write the information to. Click on **OK** to copy the information to the file.

To select which details are included in the printout:

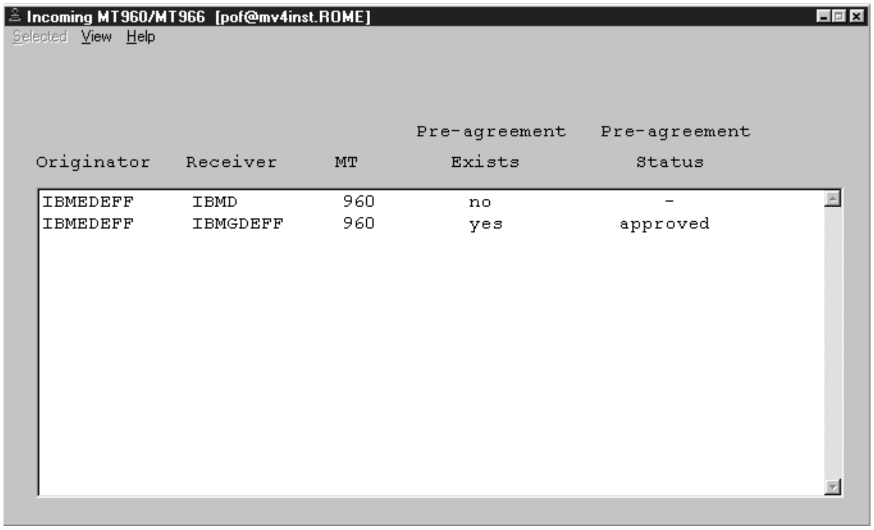
1. Select which items to include in the printout, whether the list is to be sorted by city, correspondent address, or name, and whether to sort the list in ascending or descending order.
2. To save the currently selected print options for later use, click on the **Save as default** push button.

Dealing with Incoming MT960 and MT966s

If an incoming MT960 (BKE Initiation Request) message is covered by an approved pre-agreement, it is normally dealt with for you automatically. However, if a potential correspondent sends you an MT960 without first negotiating a pre-agreement, or your pre-agreement is not approved yet, you must check the MT960 to decide whether or not you wish to proceed with key exchange. Incoming MT960 messages are routed to a special queue and handled by the **Incoming MT960/MT966** program from the **SWIFT USE UKMO** program group.

To deal with incoming MT960s:

1. Invoke the **Incoming MT960/MT966** program from the **SWIFT USE UKMO** program group. The incoming MT960s appear in the list, as shown in Figure 43 on page 71.

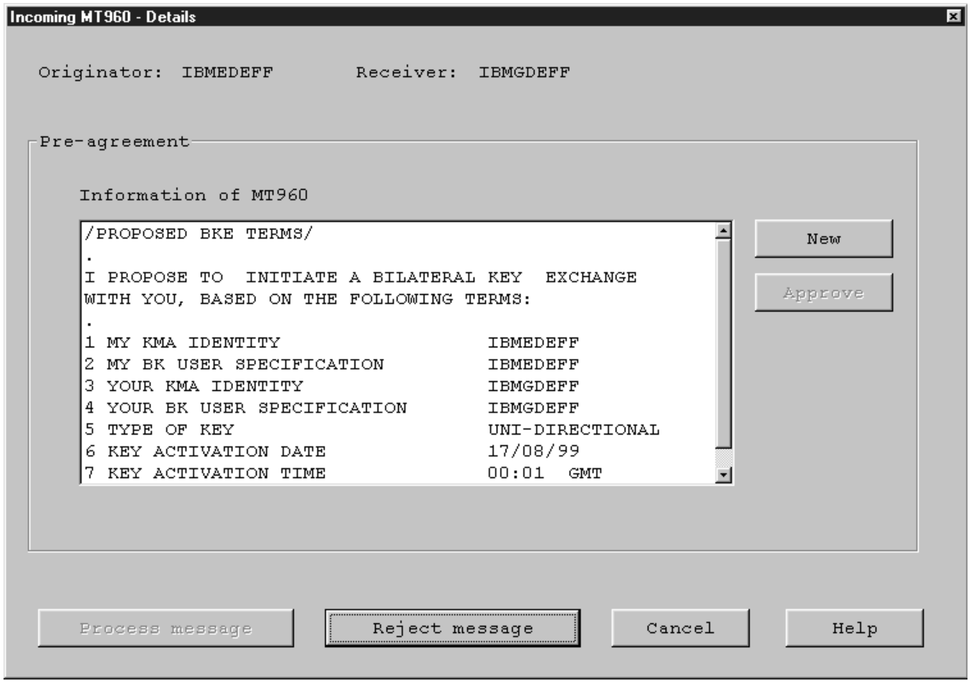


```
                                      Pre-agreement   Pre-agreement
Originator   Receiver      MT           Exists           Status

IBMEDEFF     IBMD          960          no                 -
IBMEDEFF     IBMGDEFF      960          yes             approved
```

Figure 43. The Incoming MT960/MT966 Window

2. Select the message you want to deal with.
3. Select **Open as details** from the **Selected** pull-down menu or double-click on the selected message. The Incoming MT960 - Details window appears, as shown in Figure 44.



Originator: IBMEDEFF   Receiver: IBMGDEFF

Information of MT960

```
/PROPOSED BKE TERMS/
.
I PROPOSE TO  INITIATE A BILATERAL KEY  EXCHANGE
WITH YOU, BASED ON THE FOLLOWING TERMS:
.
1 MY KMA IDENTITY                   IBMEDEFF
2 MY BK USER SPECIFICATION          IBMEDEFF
3 YOUR KMA IDENTITY                 IBMGDEFF
4 YOUR BK USER SPECIFICATION        IBMGDEFF
5 TYPE OF KEY                       UNI-DIRECTIONAL
6 KEY ACTIVATION DATE               17/08/99
7 KEY ACTIVATION TIME               00:01  GMT
```

Figure 44. The Incoming MT960 - Details Window

The following information is displayed about each message:

- The BIC identifier of the originator of the message
- The BIC identifier of the receiver of the message
- The proposed terms of the pre-agreement contained in the message. This information is optional free-format text describing the terms under which the

correspondent proposes to exchange keys. If no terms are included, the window contains the information that no pre-agreement information is included.

4. To accept the terms proposed:
   - Click on the **New** push button to create a new pre-agreement based on these terms. The originator and receiver information is automatically copied to the new pre-agreement. After creation of a pre-agreement, the **New** button is removed and an **Edit** button appears. You can edit the created pre-agreement until it is approved.
   - Click on the **Approve** push button to approve the newly created pre-agreement.

     Note: The approval of pre-agreements is only possible if the right item **Bilateral Keys - Approve pre-agreement** is assigned to the user.
   - Click on the **Process message** push button to process the MT960 message and continue the exchange as normal.
5. If you do not agree with the terms proposed by your correspondent, click on **Reject message**.

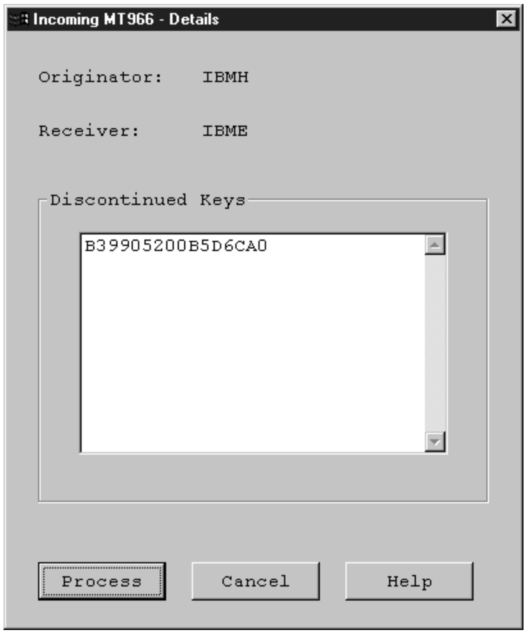
   An MT964 (BKE Protocol Error) message is sent to SWIFT You should then contact your correspondent, agree terms for the pre-agreement, and begin the normal procedure for creating a pre-agreement, as described in 'Creating a Pre-Agreement' on page 48.

To deal with an incoming MT960 message as a normal message:

1. Select the message to be processed from the list.
2. Select **Process** from the **Selected** pull-down menu.

To deal with incoming MT966s:

1. Invoke the **Incoming MT960/MT966** program from the **SWIFT USE UKMO** program group. The incoming MT966s appear in the list, as shown above in the Incoming MT960/MT966 window.
2. Select the message you want to deal with.
3. Select **Open as details** from the **Selected** pull-down menu or double-click on the selected message. The Incoming MT966 - Details window appears, as shown Figure 45 on page 73.



Originator: IBMH

Receiver: IBME

Discontinued Keys

B39905200B5D6CA0

Figure 45. The Incoming MT966 - Details Window

The following information is displayed about each message:

- The BIC identifier of the originator of the message
- The BIC identifier of the receiver of the message
- The keys to be discontinued

4. To process the MT966 message, select **Process**.

   Note: If you do not process the message, the keys are not discontinued. The MT966 message remains in the queue. If you want to delete the MT966 message without processing, use the Delete function of the Client.

To deal with an incoming MT966 message as a normal message:

1. Select the message to be processed from the list.
2. Select **Process** from the **Selected** pull-down menu.

Backing Up Bilateral Keys and Pre-Agreements

You have to make a regular backup of your bilateral keys and pre-agreements, so that in case of an emergency you still have a copy of what would otherwise be unrecoverable information. If you lose your bilateral keys and pre-agreements without having made a backup copy, you first have to prepare pre-agreements for all the correspondent relationships you need, and you must exchange bilateral keys with all your correspondents again before being able to send or receive financial messages.

You *must* back up your existing bilateral keys and pre-agreements before you generate and install a new Secure Transmission Key (STK).

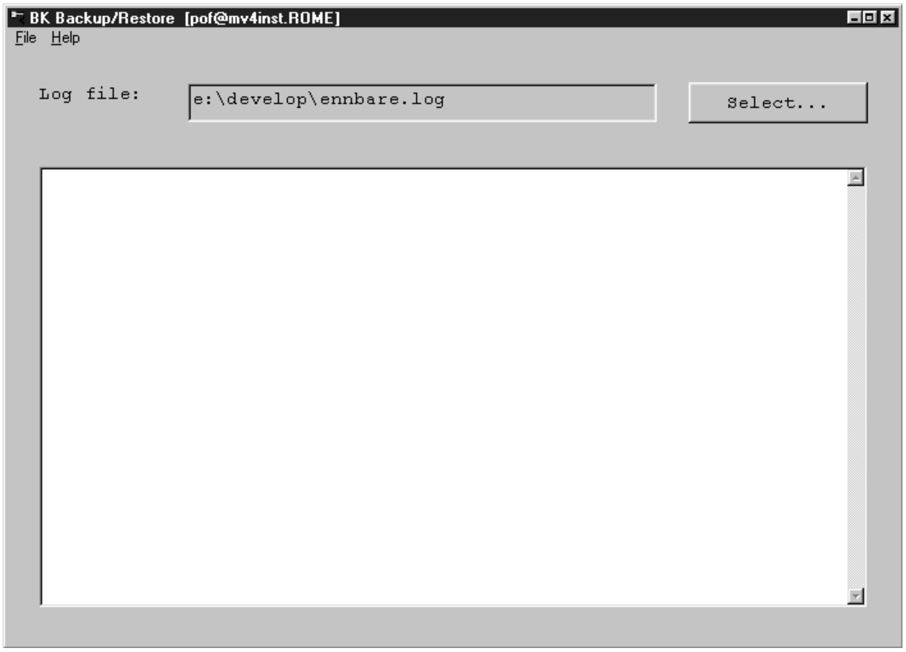
The backup process makes a copy of the bilateral keys and the pre-agreements to a file you specify.

Note: To ensure data integrity, you cannot use any other MERVA USE program in parallel with **BK Backup/Restore**.

To restore a backed-up copy of your keys or pre-agreements, do the following:

1. Invoke the **BK Backup/Restore** program from the **SWIFT USE UKMO** program group.

   The BK Backup/Restore window is displayed, as shown in Figure 46.



Log file: e:\develop\ennbare.log

Figure 46. The Backup/Restore Bilateral Keys Window

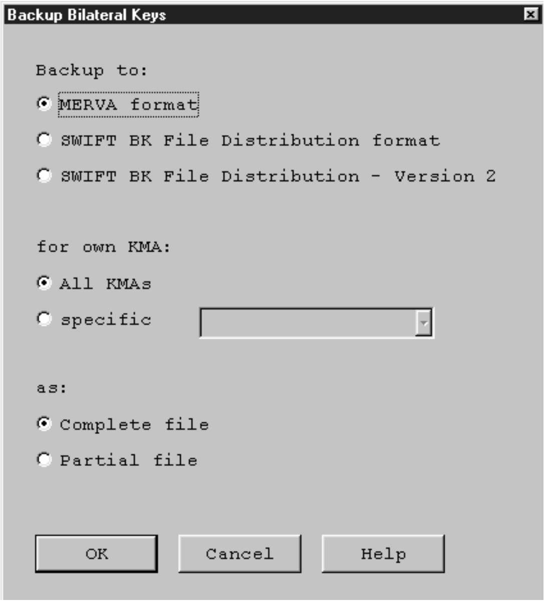
   Log File

   BK Backup/Restore writes information about keys and pre-agreements, that are being backed up or restored, to a file. By default, the name of the file is **ENNBARE.LOG**. Its name can be changed by pressing the **Select...** button. The file is overwritten each time a backup or restore is made.

   Important information also appears in the list box area of the main window. The list box also is cleared each time before a new backup or restore starts.

2. Select the **Backup** choice from the **File** pull-down menu.

   The Backup Bilateral Keys window appears, as shown in Figure 47 on page 75.


Backup Bilateral Keys

Backup to:

● MERVA format

○ SWIFT BK File Distribution format

○ SWIFT BK File Distribution - Version 2

for own KMA:

● All KMAs

○ specific

as:

● Complete file

○ Partial file

Figure 47. The Backup Bilateral Keys Window

It allows you to back up both, your authenticaton key data and your pre-agreement data in MERVA format, or your authentication key data in SWIFT format.

3. Select the format that the keys are to be backed up in:
   - MERVA format

     If MERVA format is selected for backup, all information belonging to bilateral keys is written to the file in ciphered form. This allows to restore the bilateral key information even if the STK is changed in the system.

     Note: The file is additionally protected by a password.
   - SWIFT authentication key data format

     If SWIFT authentication key data format is selected for backup, only the bilateral key information is written to the file in ciphered form, excluding STK and pre-agreement information.

     This format provides a common file structure and can be used for distribution of authentication key data between different CBT Vendors.

     The following restrictions apply to the SWIFT authentication key data format:

     a. The Secure Transmission Key (STK) on the CBT, where the BK Backup is created, and on the CBT, where the backup is restored, **must not** be different.

     b. No BK backup in SWIFT authentication key data format is possible, if there is no STK available.

     c. The SWIFT authentication key data format supports only 32-character BKs and therefore existing 16 character manual BKs are padded to 32-character BKs when written to the SWIFT BK File. This padding follows the rules given by SWIFT

        Note: A 16-character BK key that was backed up in SWIFT format appears as 32-character BK key on the CBT where the file is restored.

   d. Only correspondent relationships that contain at least one send or receive key, or an approved pre-agreement, are backed up in SWIFT format.

- ’SWIFT authentication key data Version 2’ format

  This format is an enhancement of the SWIFT authentication key data format. It also supports delta files.

  The restrictions and other specifications are equal to those of the SWIFT authentication key data format. Also see “Conversion of SWIFT Authentication Key Data Version 2 into Version 1” on page 77.

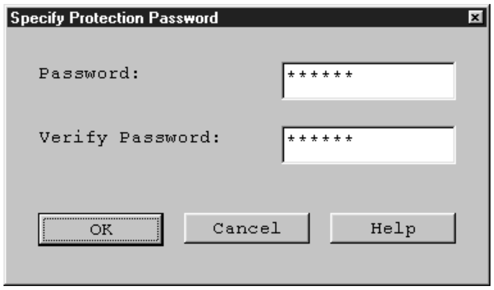
4. The KMA field allows to specify which correspondent relationships are backed up. Correspondent relationships containing the selected KMA as own KMA destination are written to the file. You can either choose to backup relationships belonging to all KMAs in the database (only available in MERVA format), or to select a specific KMA from the drop-down list. The list contains all own KMAs (without Branch code) you currently have in your authentication key database. To back up all correspondent relationships using **MERVA format**, select **ALL** from the KMA list.

   The KMA must be 8 or 11 characters long. In both cases all relationships containing this KMA with any Branch code are backed up.

5. Select the file format (complete or partial file). The file format is used to specify whether a complete backup or partial backup file is to be generated.
   - Select **Complete file** if you want to make a complete copy of the correspondent relationships belonging to the specified KMA. When you restore the file with a MERVA system, you are asked whether you want to delete all correspondent relationships belonging to the specified KMA in your database before restoring the records of the file.
   - Select **Partial file** if you want to replace and add the correspondent relationships belonging to the specified KMA when you restore it. Correspondent relationships belonging to the specified KMA in the database are **NOT** removed if they are not contained in the file.

6. Click on the **OK** push button.

7. If you choose to back up the keys in MERVA format, the Specify protection password window appears, as shown in Figure 48.



Password: ******

Verify Password: ******

Figure 48. The Specify Protection Password Window

Enter the password twice and press OK. The following password rules apply:
- Only alphanumeric characters are allowed.
- The maximum password length is 8 characters.
- The password is not displayed.

Note: The password is needed to restore the file. Therefore write down the password and keep it save.

8. The Specify file name for backup file window is displayed, as shown in Figure 49.

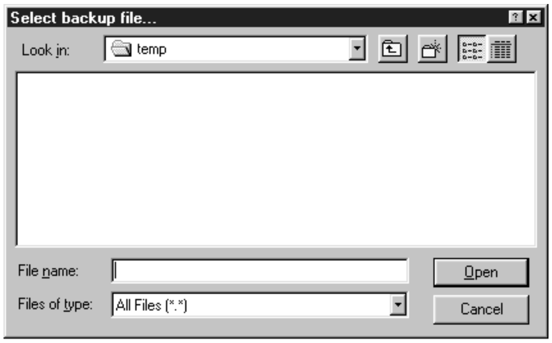
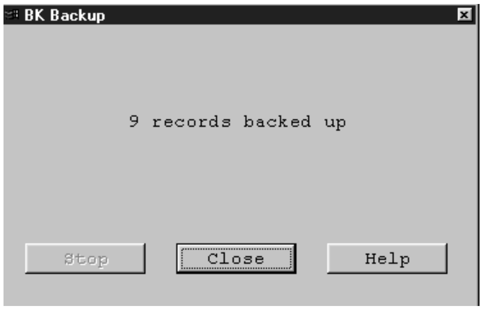


Figure 49. The Specify File Name for Backup Window

Either fill in the drive, path, and file name in the **Open filename** field, or select the drive, path, and file name from the appropriate lists.

9. Click on the **OK** push button to back up the keys.

The BK Backup window appears, as shown in Figure 50.



9 records backed up

Figure 50. The BK Backup Window

This window displays the own and correspondent destinations of the relationship currently being backed up. The backup process can be stopped by pressing the Stop button. The backup file then is incomplete and cannot be restored.

10. After the backup is made, look at the information area of the BK Backup/Restore window to check the messages there. A detailed list of the backed up relationships is contained in the log file. Use a system editor to display this log file.

Note: Check the log file each time a BK backup has been made, and keep it save until the file is no longer used.

Conversion of SWIFT Authentication Key Data Version 2 into Version 1

If you have branches that are not able to read the SWIFT authentication key data Version 2 format, you have to convert the SWIFT authentication key data Version 2 file into several SWIFT authentication key data Version 1 files.

To convert a file in SWIFT authentication key data Version 2 format into SWIFT authentication key data Version 1 format, do the following:

1. Log on to MERVA
2. In a system command prompt, enter the following command:

```
ennctsw2 OriginFile MainFile LogFile
```

where

OriginFile
is the SWIFT authentication key data Version 2 file produced by MERVA.

MainFile
is the SWIFT authentication key data Version 1 file that has to be distributed immediately.

LogFile
is the file where the logging information that is also printed on the screen, is printed to. Keep this file save. It might contain information about relationships that have to be deleted on the target system, either immediately or at a specific date.

If any relationship contains a record start date or is marked to be suspended at a scheduled date, several other files are also written. The name of those files can also be found in the log. Those files have to be restored on the target system on the date specified by their name.

A sampe log file is:

```
97-05-15 15:11:06 ENN6131I: The log file is written by File conversion tool.
97-05-15 15:11:06 ENN6118I: The log file name is ibma.log.
97-05-15 15:11:06 ENN6127I: Starting to check the file authentication code.
97-05-15 15:11:06 ENN6021I: File D:\merva\ibma.sw2 opened.
97-05-15 15:11:06 ENN6128I: The file authentication code is OK.
97-05-15 15:11:06 ENN6021I: File D:\merva\ibma.sw2 opened.
97-05-15 15:11:06 ENN6021I: File D:\merva\ibma.sw1 opened.
97-05-15 15:11:07 ENN6021I: File D:\merva\970922 opened.
97-05-15 15:11:07 ENN6021I: File D:\merva\970607 opened.
IBMADEFF -> IBMCDEFF: ENN6132I:
The record has to be manually deleted on the target system at 970712.
97-05-15 15:11:07 ENN6024I: File D:\merva\ibma.sw1 written.
97-05-15 15:11:07 ENN6024I: File D:\merva\970607 written.
97-05-15 15:11:07 ENN6024I: File D:\merva\970922 written.
```

The actions you have to perform now, are:

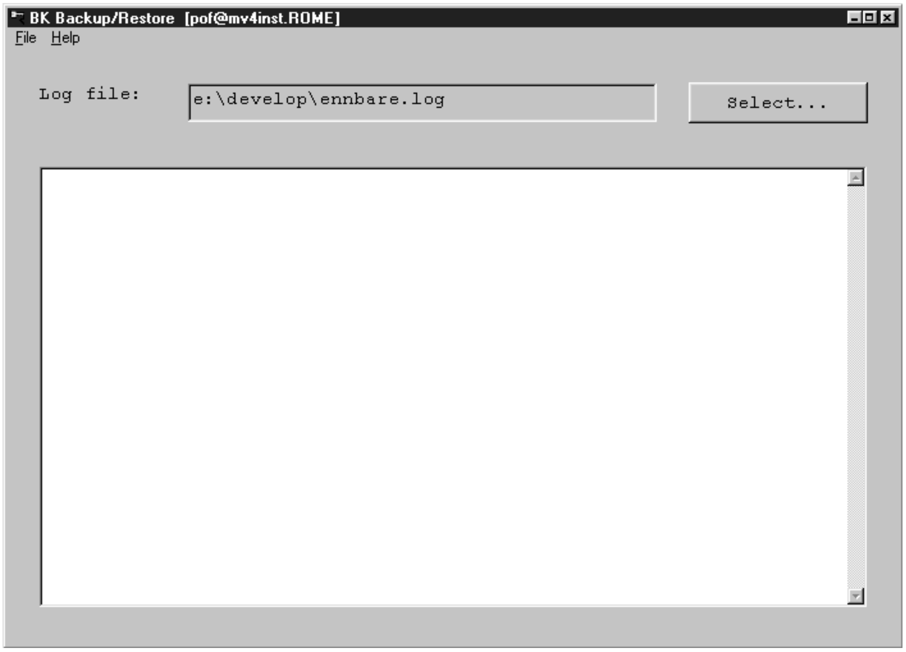
a. Restore the main file ibma.sw1 on the target system.
b. At 970607 restore the delta file 970607.
c. At 970712 delete the relationship IBMADEFF -> IBMCDEFF.
d. At 970922 restore the delta file 970922.

Restoring Bilateral Keys and Pre-Agreements

Note: To ensure data integrity, you cannot use any other MERVA program in parallel with **BK Backup/Restore**. If you restore a backed-up copy of your keys or pre-agreements, or both, existing keys or pre-agreements, or both, are overwritten. To restore a backed-up copy of your keys or pre-agreements:

1. Invoke the **BK Backup/Restore** program from the **SWIFT USE UKMO** program group.

   The Backup/Restore Bilateral Keys window is displayed, as shown in Figure 51 on page 79.



Log file: e:\develop\ennbare.log

Figure 51. The Backup/Restore Bilateral Keys Window

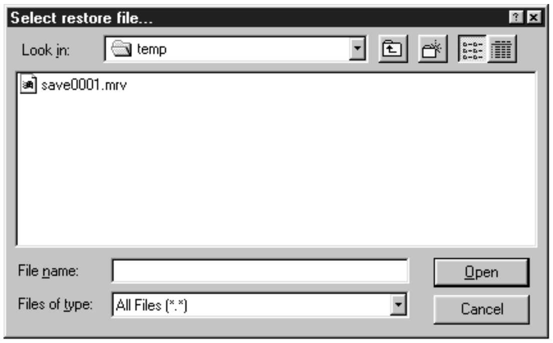
Log File

BK Backup/Restore writes information about keys and pre-agreements that are being backed up or restored to a file. By default, the name of the file is **ENNBARE.LOG**. Its name and path can be changed by pressing the **Select...** button. The file is overwritten each time a backup or restore is made.

Important information also appears in the list box area of the main window. The list box also is cleared each time before a new backup or restore starts.

2. Select the **Restore** choice from the **File** pull-down menu.

   The Select file to restore window appears, as shown in Figure 52.



Look in: temp

save0001.mrv

File name:

Files of type: All Files (*.*)

Figure 52. The Select File to Restore Window

3. Specify the file the keys are to be restored from. Either fill in the drive, path, and file name in the Open filename field, or select the drive, path, and file name from the appropriate lists.
4. Click on the **OK** push button to restore the keys. You can restore bilateral keys and pre-agreements in the following file formats:
   - MERVA ESA

- MERVA format
- SWIFT authentication key data file
- SWIFT pre-agreement data file

MERVA ESA

To restore bilateral keys from MERVA ESA do the following:

a. Use the authentication key file load program **DWSAUTLD**, specifying the ’unload’ parameter, to unload the authentication key file to a sequential data set. Refer to the *MERVA for ESA Operations Guide* for further information.

b. Transfer the created sequential file to your workstation. The file must be transferred in binary mode. Do not specify EBCDIC to ASCII translation or CRLF as record separator.

c. Use the BK Backup/Restore program to restore the keys.

MERVA Format

Use the BK Backup/Restore program of MERVA to back up and restore the authentication keys. The format protects the whole information using a password. So when restoring a MERVA backup, the Enter password window is displayed.

Note: Previously, MERVA format was called MERVA OS/2 V3.3. It was used by MERVA OS/2 V3.3 and MERVA AIX V1.2.

SWIFT Authentication Key Data

Use the BK Backup/Restore program of MERVA to back up keys for other CBTs or to restore the keys from other CBTs that generated the SWIFT authentication key data format.

SWIFT Pre-agreement Data Format

Use the BK Backup/Restore program of MERVA to restore the pre-agreements from other CBTs that generated the SWIFT pre-agreement data format.

5. Depending on the file format, the following windows are displayed:

Format	Window
MERVA format	Enter Protection Password window
SWIFT authentication key data format	SWIFT Restore Options window
MERVA ESA and SWIFT authentication key data format	Restore ESA Key File - Pre-Agreement Default Values window
SWIFT pre-agreement data format	Restore Pre-Agreements - Default Values window

The different windows and their contents are explained in the following sections.

- **MERVA format**

  When you restore a MERVA backup file, the Enter Protection Password window is displayed. The following figure shows an example of this window.

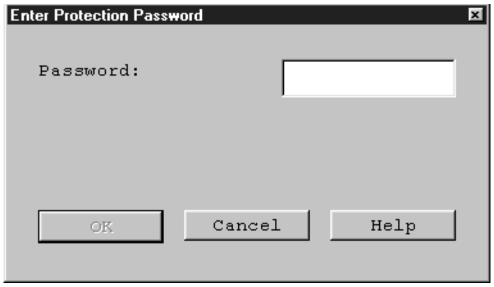
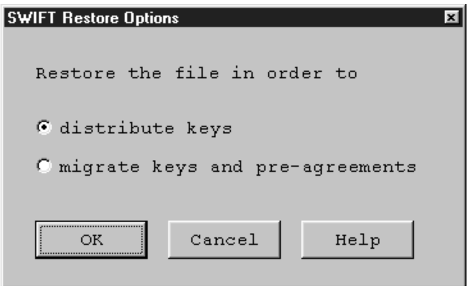


Figure 53. Enter Protection Password Window

Enter the password specified when creating the file.

- **SWIFT authentication key data format**

  The SWIFT Restore Options window is displayed as shown in Figure 54.



Restore the file in order to

● distribute keys

○ migrate keys and pre-agreements

Figure 54. The SWIFT Restore Options Window

This window offers two scenarios for restoring a SWIFT authentication key data file:

a. Distribute keys: The workstation only uses the keys contained in the file for authentication, but it is not the KMA for those relationships and no BKE takes place on this workstation for the relationships.

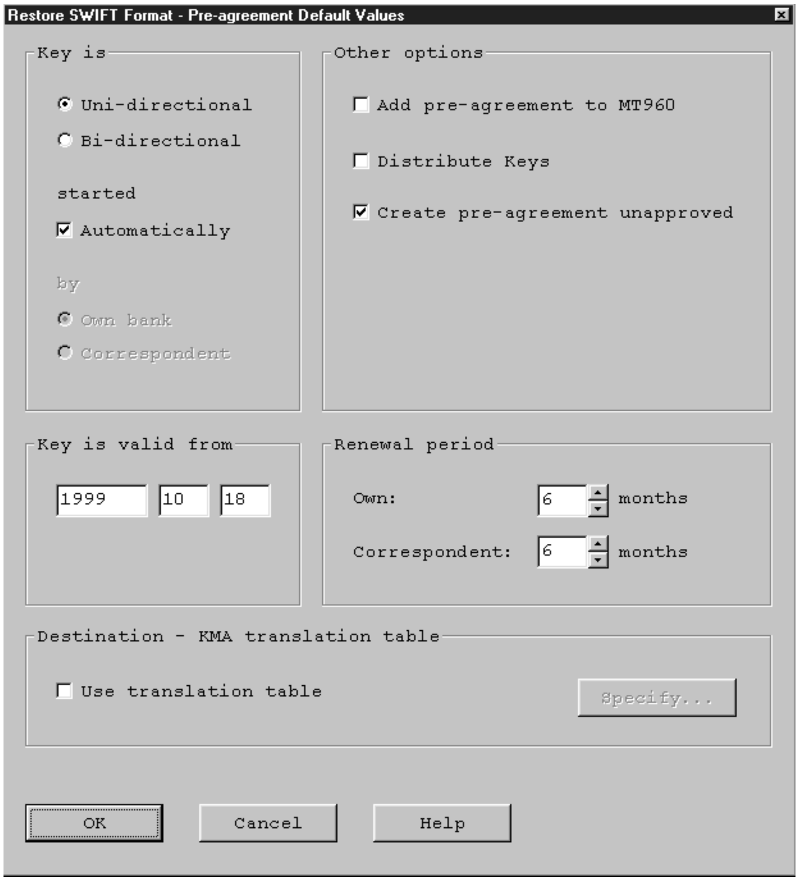
   Then the content of the default pre-agreements is not relevant, and the restore of the file starts immediately.

b. Migrate keys and pre-agreements: The workstation acts as KMA and performs BKE for relationships contained in the file. Then additional defaults have to be specified for creation of default pre-agreements, when pre-agreements for relationships in the file do not exist in the local database.

   The Restore SWIFT format - Pre-agreement default values window is displayed as shown in Figure 55 on page 82.

- **MERVA ESA and SWIFT authentication key data format**

  The Restore ESA key file - Pre-agreement default values window is displayed, as shown in Figure 55 on page 82.



Restore SWIFT Format - Pre-agreement Default Values

Key is

● Uni-directional

○ Bi-directional

started

☑ Automatically

by

● Own bank

○ Correspondent

Other options

□ Add pre-agreement to MT960

□ Distribute Keys

☑ Create pre-agreement unapproved

Key is valid from

1999 10 18

Renewal period

Own: 6 months

Correspondent: 6 months

Destination - KMA translation table

□ Use translation table

Figure 55. The Restore ESA Key File - Pre-Agreement Default Values Window

Note: When restoring a SWIFT authentication key data file, the window is named 'Restore SWIFT format - Pre-agreement default values'.

The values are used to create new pre-agreement information if the correspondent relationship for specific key data does not exist in the database.

Note: Pre-agreement information is only created if there is no information available for the specific correspondent relationship.

Key is Indicates whether the key is:

- Uni-directional or bi-directional
- Started automatically by your own bank or the correspondent bank
- Exchanged manually or via BKE

If the correspondent relationship contains no valid send or bi-directional key, then the key type is set to the default shown here. Otherwise the information is taken from the last send or bi-directional key. The default exchange technology is BKE. The **automatic start** option is always taken from the default values.

'Key is valid from' date

If at the time the next key is generated this date is in the:

- Future, the next key that is generated will be effective from this date.
- Past, and no keys already exist, the next key that is generated will be effective from the (then) current date.
- Past, and at least one key already exists, the next key that is generated will be effective from the date of the current key plus the renewal period.

Note: This date is the date in the Greenwich Mean Time (GMT) time zone.

Key renewal periods
The key renewal periods are used to calculate the expected end date of the last key if the restore file is a SWIFT authentication key data file, and to set the corresponding values in the pre-agreement.

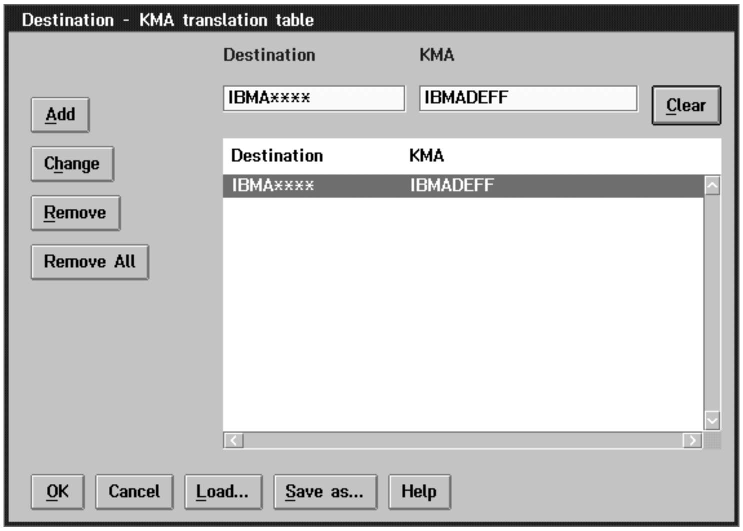
Add pre-agreement to MT960
If this check box is marked, the MT960 contains the pre-agreement data in a free-format text field.

Distribute Keys
If this check box is marked, all changes regarding keys of this correspondent relationship are automatically distributed to other MERVA systems via MERVA Link.

Create pre-agreement approved
If this check box is not marked, all pre-agreements that are created have to be approved in a separate step. **We recommend not to mark this box**, as pre-agreement information that is automatically generated has to be reviewed to ensure correct BKE processing.

It is also a method to control the amount of BKEs started automatically after a migration. BKE is only started for correspondent relationships with approved pre-agreements. With the **Bilateral Key Exchange** program contained in the UKMO folder pre-agreements of multiple correspondent relationships can be approved in one step.

Use translation table
If this check box is marked, you can define and use a translation table where you can specify KMAs for generic destinations and destinations where KMA and destination do not match exactly. If you press the **Specify...** button, the Destination - KMA translation table window is displayed, as shown in Figure 56 on page 84.



Destination - KMA translation table

Destination: IBMA****

KMA: IBMADEFF

Destination	KMA
IBMA****	IBMADEFF

Figure 56. The Destination - KMA Translation Table Window

Use the **Destination - KMA translation table** window to specify KMAs for specific destinations.

The function of the Destination - KMA translation table is the following:

The SWIFT authentication key data format does not conain a KMA for correspondent destinations, and the MERVA ESA format neither contains a KMA for the own destionation, nor for the correspondent destination. The destination/KMA table contains pairs of destinations and KMAs. So, when restoring a backup file from MERVA ESA, the table is searched for both, own destination and correspondent destination. When restoring a backup file in SWIFT authentication key data format, the table is only searched for the correspondent destination.

The process of finding a KMA for a destination that is used for MERVA ESA files and SWIFT authentication key data files, works in the following way:

a. First the Destination - KMA translation table is searched for an entry, where the destination contained in the corresponding relationship exactly matches the destination of the entry.

b. If no entry is found, the destination of the correspondent relationship itself is examined. If it is unique, it is used.

c. If the destination is not unique, the table is searched for an entry, where the destination of the correspondent relationship matches a generic destination of the entry.

d. If no matching destination has been found, an error message is written to the log file. If the backup file is in MERVA ESA format, the relationship is restored anyhow, if it is in SWIFT authentication key data format, the relationship is not restored.

Example:

```
The restore file is a MERVA ESA file
and contains the relationships

  own         correspondent
  IBMCDEFF    IBMDDEFF
  IBMC****    IBMD****
```

```
The KMA table contains

  destination KMA
  IBMC****    IBMCDEFF
  IBMCDE**    IBMCDEAA
  IBMD****    IBMDDEFF
  IBMDDE**    IBMDDEFF
  IBMDDEFF    IBMDDEAA

The result in the database is
  own dest  own KMA      corr dest  corr KMA
  IBMCDEFF  IBMCDEFF     IBMDDEFF   IBMDDEAA
  IBMC****  IBMCDEFF     IBMD****   IBMDDEFF
```

There are two different ways to specify a Destination - KMA translation table:

– Use the panel

  Enter the destination and its appropriate KMA in the entry fields. If the entries are correct (the first 4 characters have to match and the length or destination has to be 8 characters, the KMA has to be 8 or 11 characters long), the **Add** button is enabled and you can add the pair to the list of table entries.

  If you select a pair of destination/KMA in the container, it appears in the entry fields, and you can edit it. If you then press the **Change** button, the changes are adapted to the translation table list.

  To **remove** an entry from the list, just select it in the table and press the **Remove** button. The **Remove All** button removes all pairs from the list.

  The **Clear** button clears the contents of the entry fields.

  If you want to use the list you have specified before several times, save it using the **Save As...** button. A file dialog is displayed where you can choose a file name for the file. You can afterwards load that file using the **Load...** button.

– Use a file

  You can use a normal text editor to create a file that contains a pair of destination and KMA in each line, separated by one or more blanks like in the following example:

```
IBMCDEFF  IBMCDEFFKMA
IBMDDEFF  IBMDDEFFKMA
IBMCDE**  IBMCDEFFKM2
IBMC****  IBMCDEFF
```

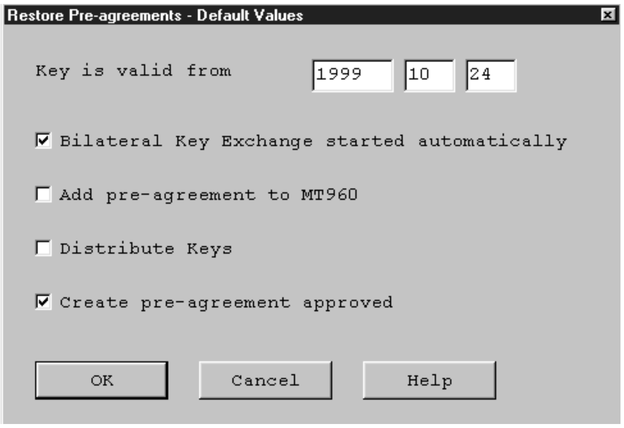
  The destination must be 8 characters long, the KMA length must either be 8 or 11 characters.

  The file then can be loaded by pressing the **Load...** button, which displays a file dialog where you have to specify the file name containing the translation table.

Note: The list is not cleared automatically when loading a file.

- **SWIFT pre-agreement data format**

  The Restore pre-agreements - default values window is displayed, as shown in Figure 57 on page 86.


Restore Pre-agreements - Default Values

Key is valid from 1999 10 24

☑ Bilateral Key Exchange started automatically

□ Add pre-agreement to MT960

□ Distribute Keys

☑ Create pre-agreement approved

Figure 57. Restore Pre-Agreements - Default Values Window

It is used to set default values that are used for the restore of a file in SWIFT pre-agreement data format. The values are used if either the data is not contained in the input file because it is specific for MERVA, or it is just not available.

Note: Pre-agreement information is only added to the database, if there is no information available for the specific correspondent relationship!

'Key is valid from' date
If at the time the next key is generated this date is in the:

- Future, the next key that is generated will be effective from this date.
- Past, and no keys already exist, the next key that is generated will be effective from the (then) current date.
- Past, and at least one key already exists, the next key that is generated will be effective from the date of the current key plus the renewal period.

Note: This date is the date in the Greenwich Mean Time (GMT) time zone.

Add pre-agreement to MT960
If this check box is marked, the MT960 contains the pre-agreement data in a free-format text field.

Bilateral Key Exchange started automatically
If this check box is marked, BKE is started automatically. This value is only used if not specified in the SWIFT file.

Distribute Keys
If this check box is marked, all changes regarding keys of this correspondent relationship are automatically distributed to MERVA systems via MERVA Link.

Create pre-agreement approved
If this check box is not marked, all pre-agreements that are created have to be approved in a separate step. **We recommend not to mark this box**, as pre-agreement information that is automatically generated has to be reviewed to ensure correct BKE processing. It is also a method to control the amount of BKEs started automatically after a migration.

BKE is only started for correspondent relationships with approved pre-agreements. With the **BK Backup/Restore** program contained in the UKMO program group, pre-agreements of multiple correspondent relationships can be approved in one step.

Note: If the renewal periods in the file are not applicable, they are set to the SWIFT default value of 6 months.

The key exchange technology is set to bilateral key exchange for all pre-agreements.

Chapter 4. Tasks of the User

The user performs the following administrative tasks (a USER card is required for these procedures):

- **Associating logical terminals with card readers**

  Because session keys are specific to a logical terminal (LT), the user must specify the association between card reader and LT before generating session keys.
- **Changing the access technology flag**

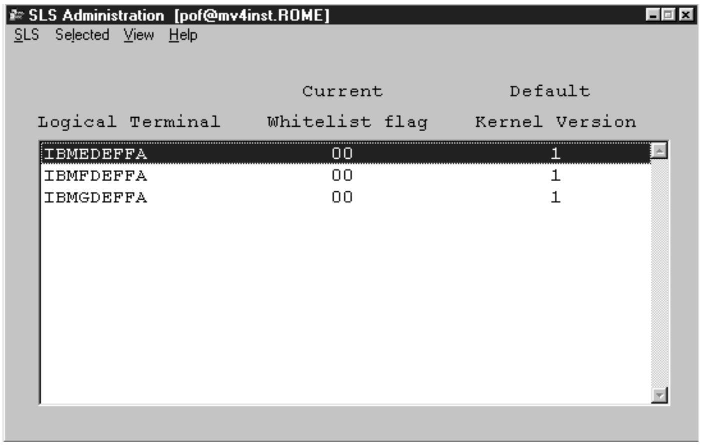
  The user is responsible for informing SWIFT of the type of access technology (paper tables or cards) that is used to access SWIFT services.
- **Pregenerating session keys**

  A session key is a number that must be entered for each LOGIN and SELECT request. MERVA provides a facility to pregenerate these session keys and store them in the system ready for use by the LOGIN and SELECT processes or for distribution to other MERVA systems. The user is responsible for pregenerating these session keys.

Associating Logical Terminals with Card Readers

Because session keys are specific to a logical terminal (LT), you must specify the association between card reader and LT before generating session keys. To do this:

1. Invoke the **SWIFT SLS Administration** program from the **Communication** program group. The SLS Administration window appears, as shown in Figure 58.



Logical Terminal	Current Whitelist flag	Default Kernel Version
IBMEDEFFA	00	1
IBMFDEFFA	00	1
IBMGDEFFA	00	1

Figure 58. The SLS Administration Window

   The Logical Terminal is listed on the SLS Administration window, together with the current whitelist flag value and default kernel version.

   The **Default Kernel Version** is the value of the kernel version stored in the MERVA database. This value is used for session key requests (coming from MERVA ESA, for example) where no specific kernel version is specified.

2. Select **New LT** from the **SLS** pull-down menu.

3. Each card reader can support any number of LTs. An LT, however, must be assigned to a single card reader only. The list only includes those LTs that have not been assigned to another card reader.

   Enter an LT and select the card reader to use to generate session keys for this LT.

   The serial number (ID) of the selected card reader is displayed.
4. Click on the **OK** push button.

Changing the Access Technology Flag

The type of technology used to make a LOGIN or SELECT request to the SWIFT network is known as the access technology. The two types of access technology available are:

- Paper tables
- Cards (ICCs)

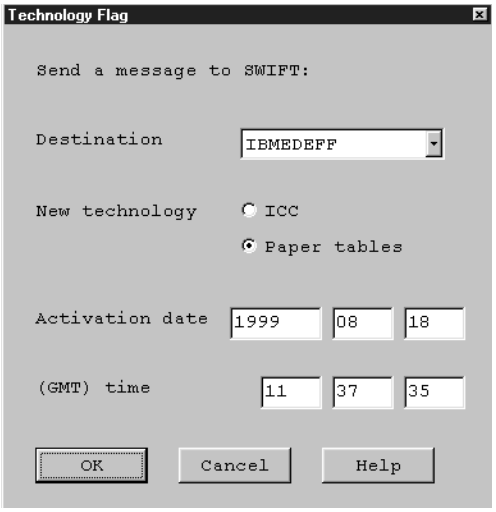
The type of access technology you use is stored on the workstation in a parameter called the technology flag. The technology flag applies to all LTs of a destination.To verify a LOGIN or SELECT request correctly, SWIFT must be aware of the access technology used by your workstation. Therefore, you must inform SWIFT of changes to the technology flag. This is done by sending an MT090 (Change Access Technology Request) message to SWIFT specifying the date and time when the change of technology is to take place.

It is your responsibility to change the technology flag at this date and time. If, for example, you inform SWIFT that a change of technology is to take place in one week's time, you must remember to change the technology flag at that time.

To change the technology flag:

1. Select **Change Technology Flag** from the **SLS** pull-down menu.

   The Technology Flag window appears, as shown in Figure 59.



Technology Flag

Send a message to SWIFT:

Destination: IBMEDEFF

New technology: ○ ICC ● Paper tables

Activation date: 1999 08 18

(GMT) time: 11 37 35

Figure 59. The Technology Flag Window

2. Click on the radio button for **ICC** or **Paper tables** to select the technology required. The initial value is not the value set the last time the window was displayed.

3. Specify the date and time from when the chosen technology is to be used.
4. Select the destination to which the change of technology applies.
5. Click on the **OK** push button. You are asked to confirm that you want to send an MT090 (Change Access Technology Request) message and send it to SWIFT

SWIFT sends an MT092 to confirm the change of access technology.

Pregenerating Session Keys

You pregenerate session keys for an LT so that the LT can perform a LOGIN or SELECT to the SWIFT network without needing to continuously request session keys from a card reader.

You can pregenerate maximum 9999 session keys at one time. For example, you could pregenerate as many session keys as you need for a week, a month, or a year. Alternatively, you could pregenerate 1000 session keys for one set of whitelist flag, kernel version, and set number of cards used in the SLS service.

You can pregenerate session keys for subsequent use on the local MERVA system or for distribution to another MERVA system, such as MERVA ESA.

To pregenerate session keys:

1. Invoke the **SWIFT SLS Administration** program from the **Communication** program group.
2. Insert the USER card into the card reader.
3. Select the logical terminal to generate session keys for.
4. Select **Key pregeneration for local MERVA system** from the **Selected** pull-down menu to generate keys for the local system. Select **Key pregeneration for distribution** to generate and distribute keys to another MERVA system.
5. If necessary, update the kernel version, whitelist flag, and card set values.

   Each of the ICC kernel versions held on a USER card can only be used to generate LSNs or SSNs between 0001 and 9999. Once the maximum number has been reached, you must change the kernel version in use.
6. The **Current number** and **Remaining** fields show the values of the current LSN or SSN and the number of pregenerated LSNs or SSNs remaining.
7. In the **Generate** field, enter the numbers of LSNs and SSNs to create.
8. Click on the **OK** or **Generate** push button, respectively. The session keys are generated for the previously selected logical terminal and stored in the MERVA database.

To send a range of session keys to another MERVA system:

1. Display the Pregeneration of Session Keys for Distribution window, as described previously.
2. In the **Distribute** fields, select a range of LSNs or SSNs to send to the other system.

   The range you select must lie within the range of LSNs or SSNs available.
3. Click on the **Send** push button.

Deleting Session Keys

You can delete session keys, or specific ranges of session keys, that you have previously generated but no longer require.

To delete session keys:

1. Invoke the **SWIFT SLS Administration** program from the **Communication** program group.
2. Select the logical terminal to delete the session keys for from the list.
3. Select **Delete session keys** from the **Selected** pull-down menu.
4. The current ranges of pregenerated LSNs and SSNs are displayed.

   Choose the session keys to delete:

   - You can only delete session keys from within the currently available range.
   - The range of keys can either start from the beginning of the available range or end at the top of the available range, but cannot be from the middle of the available range.
   - For session keys on the local system, you should always delete the entire range of available keys.
5. Click on the **OK** push button.
6. On the message window that appears, click on the **Yes** push button to confirm deletion.

Deleting LTs

You can delete LTs that you have previously defined but no longer require:

1. Invoke the **SWIFT SLS Administration** program from the **Communication** program group.
2. Select the LT to delete.
3. Select **Delete LT** from the **Selected** pull-down menu.
4. Click on **Delete** to confirm deletion of the LT.

   When you delete an LT, all pregenerated session keys associated with the LT are also deleted.

Appendix A. Upgraded Secure Card Reader

As computers have become more powerful, cryptographic algorithms have become more vulnerable. Also, to ensure the integrity of cryptographic algorithms, their public keys should be changed from time to time. For these reasons, the following new functionality has been added to secure card readers (SCRs):

- An SCR can now use a 1024-bit public key of a customer application (PKA) in addition to a 512-bit PKA. Doubling the size of a PKA drastically increases the effort required to decipher it.
- An SCR can now simultaneously store and use two versions of each certificate value (CV). This provides SWIFT with a way to change its public key (PKS) while still ensuring that all banks, including those that have not yet requested and installed new CVs based on this new PKS, can still exchange bilateral keys with each other.

An SCR that has been upgraded with this functionality has a label on it with the text **Upgraded SCR**.

Increasing a PKS Version

Because CVs are generated using the PKS (which is managed by SWIFT), introducing a new PKS means that new and different CVs are needed as well. Not all SCRs will be able to implement the new CVs simultaneously, so there will always be a period during which some SCRs will have CVs based on the new version and others will have CVs based on the older version. To allow SCRs to continue to work together, SCRs have been given the capability to simultaneously store and use two versions of each CV: a lower version and a higher version. These CV versions correspond directly to the PKS versions.

Holding two versions of CVs simultaneously lets an SCR with CVs based on the latest PKS versions continue to work with an SCR that has not yet implemented the latest versions. For example, currently any two SCRs can both provide CVs based on PKS version 0, even if one has both versions 0 and 1 available, and other only version 0.

Only after all customers have upgraded their SCRs and software and have installed CV versions 0 and 1 can SWIFT implement PKS version 2 (and generate version 2 CVs). An SCR can store only two CV versions at a time, so when SWIFT switches to PKS version 2, all version 0 CVs must be manually deleted from each SCR (these will no longer be needed, as all SCRs will have version 1 CVs available) before the version 2 CVs can be stored. After this is done for every SCR in the network, SWIFT can consider introducing PKS version 3.

The size of a PKA or CV depends on the CV version:

Version	PKA Size	CV Size
0	512 bit	640 bit
1 or higher	1024 bit	1152 bit

When you use the latest MERVA software to request a new CV, you are automatically sent two CVs: one for each of the two versions currently in use. Be sure you have the SCR connected to the system when the CVs arrive from SWIFT (MT087), because only then will both versions be stored, and only them can you be sure that you will be able to exchange bilateral keys with all correspondents.

Bilateral Key Exchange (BKE) Processing

Because SWIFT will not introduce a new PKS version until all banks have implemented the current ones, there are currently only 4 possible scenarios:

- Scenario 1: Your and the correspondent’s SCRs have available only CVs of version 0.
- Scenario 2: Your and the correspondent’s SCRs have available CVs of versions 0 and 1.
- Scenario 3: Your SCR has available only CVs of version 0; the correspondent’s SCR has available CVs of versions 0 and 1.
- Scenario 4: Your SCR has available CVs of versions 0 and 1; the correspondent’s SCR has available only CVs of version 0.

During bilateral key exchange, your bank contacts a correspondent and passes to it (in an MT960) a CV of its highest CV version:

- If the correspondent’s SCR has that version available (Scenarios 1, 2, and 3 above), the correspondent returns (in an MT961) a CV of the same version, and communication can commence.
- If not (Scenario 4 above), the correspondent returns its SCR’s highest version (in this case version 0). When this happens, your SCR tries again using its lower version (version 0). This version is available to both SCRs, and communication can commence.

Old and New SCRs Connected to the Same System

If an upgraded SCR is connected to a system to which an SCR is already connected, there is no need to remove the old SCR. However, note that MERVA cannot handle two SCRs that have CVs stored for the same destination.

Secure Transmission Key (STK) Processing

The STK processing is unchanged for the upgraded SCR. If you have acquired an upgraded SCR, install the old STK into the new SCR. If you connect more than one SCR to the same MERVA system, they must all have the same STK.

The ICC Kernel

MERVA does not use the ICC kernel stored in the card reader; instead, it uses data stored in its database or entered from the LOGIN panel. For this reason, you do not need to update the kernel version in the new card reader.

Installation Instructions

Unless told otherwise, MERVA assumes the lowest CV version is 0. If your SCR’s lowest CV version is greater than 0, you must do the following:

1. In Windows NT, from the Start button, select Settings, then Control Panel.
2. From the control panel, select System.
3. From the System notebook, select the Environment page.

4. On the Environment page, in the list of system variables, select any system variable.
5. In the Variable field, overtype the name shown with ENM_VERSION_LOW.
6. In the Value field, enter the lowest version as a three digit number; pad it with leading zeroes if necessary (for example 001).
7. Reboot Windows NT.
8. Start MERVA. MERVA reads this system variable, and writes the value into the database.
9. In Windows NT, from the Start button, select Settings, then Control Panel.
10. From the control panel, select System.
11. From the System notebook, select the Environment page.
12. On the Environment page, in the list of system variables, select the system variable ENM_VERSION_LOW (the one you just created).
13. Press the Delete button.

If SWIFT later increases the version, MERVA will recognize this and adjust the level automatically. It is important that you complete all of the above steps. If you do not delete the system variable ENM_VERSION_LOW, this will result in an error when SWIFT increases the PKS version.

Appendix B. Automatic Distribution of BK Data via MT999

Here are the conditions described, when an MT999 message used to update the authenticator-key file on another MERVA system is automatically generated. In general, automatic generation of distribution requests (MT999) is only activated if in the pre-agreement for that correspondent relationship the ’Distribute Keys’ flag is set on.

A manual distribution of BKs can be done at any time by selecting the record from the BKE list of Correspondents and clicking on ’Distribute’ from the ’Selected’ pull down.

An MT999 is automatically generated under one of the following conditions:

1. A BKE key has successfully been exchanged. The status of the new key is **Current** or **Future**. There is no open protocol (for example: Await...) for this key.
2. The ’Save’ button has been pressed on the Correspondent - Bilateral Keys panel and one of the following actions were performed:
   - A manual key has been entered or changed
   - ’Change Date’ has been performed
   - BKE counter(s) have been incremented or reset
   - A key has been discontinued and the status of the key in MERVA is ’Await MT967’. The MT999, however, contains the key entry with status ’Discontinued’. Note that no other MT999 is sent afterwards, even if the status of that key in MERVA is changed later to ’Timeout MT967’, ’Discontinue error’ or ’Discontinued’. From the moment when the MT966 is generated, that key is handled as discontinued key in MERVA.
3. An existing key has been overwritten by a new BKE start. (status ’Await MT961’ or ’Await MT962’). An MT999 with a blank entry for that key is generated, to indicate that this key does not exist anymore. It will eventually be overwritten by a new MT999 when the key is successfully exchanged.
4. The start of a new BKE causes the ’dropping’ of an old previous key. All following keys are shifted, so that an entry is available for the new key to be generated. The MT999 contains the keys and the last entries (3 and 6 for bi-directional) are blank. These will eventually be overwritten by a new MT999 when the key has been exchanged successfully.
5. A Discontinue Request MT966 is received and processed by the incoming MT960/MT966 function or USE Background process. The MT999 contains an appropriate entry marked as ’Discontinued’. Even if the MT966 is responded by an MT964, the key is handled as discontinued and the status is ’Discontinue error’.
6. An MT964 follows an MT963 on the initiator side. On the sending side of the MT963 this key was already current or future and an MT999 has already been sent with the new key.
7. An MT964 follows an MT963 on the responer side. If the MT964 arrives on a current key, the status of the key is set to ’Discontinued’ and an MT999 is generated with that status.

   If the MT964 arrives on a future key, the status is set to ’Error detected by Corresp.’ and an MT999 is generated with the appropriate entry(ies) being blank.

8. The whole relationship is deleted. An MT999 with a delete request for that record is generated.
9. A pre-agreement of a relationship that contains keys is updated and approved with any change to the following fields:
   - Start date
   - End date
   - Exclusion
10. A relationship is approved as *suspended* or as *reactivated*.

The BKE context states
- Await..
- Error detected...
- Timeout..

are reported as blank entry for the appropriate key in the MT999.

Notes:
1. A relationship that does not have any keys but that has set **Start date**, **End date**, or **Exclusion**, is not distributed automatically. This is also valid if the flag **Distribution key** is set in the pre-agreement. You must start the distribution of a relationship of this kind manually.
2. MERVA cannot ensure that the order of distribution messages received by another MERVA system is identical to the order of distribution messages sent by another MERVA system within a short period of time. For example, if a record is updated twice, it is not specified which MT999 arrives first.

Appendix C. Working with Several KMAs

Institutions with more than one bank code often have several KMAs; one for each bank code. Each KMA is strictly responsible for only those BKE records that belong to a specific bank code.

To ensure that, this KMA **must not** have the **Bilateral Keys – All** right defined in User Maintenance, but must have an LT defined in the user record. The first eight characters of this LT define, which BKE records you can work with.

The BKE Window

In the BKE window (Figure 28 on page 47) such a restricted KMA only sees those BKE records where the first eight characters of the LT (Logical Terminal) in the user record are equal to the first eight characters of the own KMA defined in the BKE record.

Example:

```
LT in the User Record:  IBMCDEFFA
Following BKE records are in the database:
  Own destination     Own KMA
    IBMCDEFF          IBMCDEFF
    IBMC****          IBMCDEFF
    IBMCBE**          IBMCDEFF
    IBMCIT**          IBMCITFF  <-- not be seen on the BKE window
    IBMDDEFF          IBMDDEFF  <-- not be seen on the BKE window
```

Only the first three records are displayed in the BKE window. The KMA can only select from this list and, subsequently, can only look at Approve, Print, Start BKE, Delete, and Send to distribution records from that list.

Delete All and Print All

The Delete All and Print All functions from the Relationship pull-down delete or print all those records in the database that belong to this KMA.

Create or Update Pre-Agreement

The restricted KMA can only create pre-agreements where the first eight characters of the own KMA are equal to the first eight characters of the LT defined in the User Maintenance. If a branch code is present in the LT, it is displayed as default in the own KMA field. The branch code, however, can be chosen by the KMA. The branch code defined in the 'Own KMA' is taken for all BKE messages MT960 to MT967. If no branch code is defined in the 'Own KMA', the branch code for the BKE messages is taken from the 'Emitting LT' of the 'Use message header' definition of the customizer.

Example:

```
                      LT in the User Record:  IBMCDEFFAXXX
Emitting LT in Use Message Header database:  IBMCDEFFAYYY
                                   OWN KMA:  IBMCDEFFAZZZ
```

BKE messages MT960 to MT967 have IBMCDEFFAZZZ as emitting LT. All other USE messages like MT090 have IBMCDEFFAYYY as emitting LT.

Bilateral Keys Backup/Restore

Even if a branch code is given in the 'Backup for Own KMA' field all records with the first eight characters of the own KMA equal to the first eight characters of the entered KMA are part of the file.

Example:

```
LT in the User Record:  IBMCDEFFAXXX
   Backup for own KMA:  IBMCDEFFYYY
Following BKE records are in the database:
  Own destination     Own KMA
    IBMCDEFF          IBMCDEFF
    IBMC****          IBMCDEFFZZZ
    IBMCBE**          IBMCDEFFYYY
    IBMCIT**          IBMCITFF  <-- will not be backed up
    IBMDDEFF          IBMDDEFF  <-- will not be backed up
```

If 'ALL' has been specified for Backup to MERVA format, all five records will be included in the file.

Appendix D. Stopping the Usage of Bilateral Keys

The following table summarizes the identified reasons why the usage of bilateral keys can be stopped.

Reason	Applicable to	Desired Result	Mechanism
Security breach	Single keys, single relationships, all relationships	Immediate stop of authenticated traffic and no future use of affected key(s)	Discontinuation process; see "Discontinuing a Key in an Emergency" on page 60.
Change / Termination of correspondent relationship, Close of business	One or more relationships	Planned stop of authenticated traffic and deactivation of affected key(s)	1. Scheduled suspend; see "Suspension and Re-Activation of Relationships" on page 65. 2. Delete the relationship; see "Deleting a Correspondent" on page 63.
Disaster, war, blockage, embargo, temporary unavailability, business suspension by regulatory body	One or more relationships	Temporary deactivation of correspondent relationship.	Scheduled suspend; see "Suspension and Re-Activation of Relationships" on page 65.
Cancellation of exchanged key	Single key	Key is unusable	• New BKE, so key is overwritten (preferred mechanism) • Discontinuation process, if counterpart does not want to re-initiate BKE to enforce the counterpart to re-initiate BKE. See "Discontinuing a Key in an Emergency" on page 60.

Appendix E. Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Deutschland
Informationssysteme GmbH
Department 3982
Pascalstrasse 100

70569 Stuttgart
Germany

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement or any equivalent agreement between us.

The following paragraph does apply to the US only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States, other countries, or both:

- Advanced Peer-to-Peer Networking
- AIX
- APPN
- C/370
- CICS
- CICS/ESA
- CICS/MVS
- CICS/VSE
- DB2
- DB2 Universal Database
- Distributed Relational Database Architecture
- DRDA
- IBM
- IMS/ESA
- Language Environment
- MQSeries

- MVS
- MVS/ESA
- MVS/XA
- OS/2
- OS/390
- RACF
- VisualAge
- VSE/ESA
- VTAM

Workstation (AWS) and Directory Services Application (DSA) are trademarks of S.W.I.F.T., La Hulpe in Belgium.

Pentium is a trademark of Intel Corporation.

PC Direct is a trademark of Ziff Communications Company in the United States, other countries, or both, and is used by IBM Corporation under license.

C-bus is a trademark of Corollary, Inc. in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

Glossary of Terms and Abbreviations

This glossary defines terms as they are used in this book. If you do not find the terms you are looking for, refer to the *IBM Dictionary of Computing*, New York: McGraw-Hill, and the *S.W.I.F.T. User Handbook*.

A

ACB. Access method control block.

ACC. MERVA Link USS application control command application. It provides a means of operating MERVA Link USS in USS shell and MVS batch environments.

Access method control block (ACB). A control block that links an application program to VSAM or VTAM.

ACD. MERVA Link USS application control daemon.

ACT. MERVA Link USS application control table.

address. See *SWIFT address*.

address expansion. The process by which the full name of a financial institution is obtained using the SWIFT address, telex correspondent's address, or a nickname.

AMPDU. Application message protocol data unit, which is defined in the MERVA Link P1 protocol, and consists of an envelope and its content.

answerback. In telex, the response from the dialed correspondent to the WHO R U signal.

answerback code. A group of up to 6 letters following or contained in the answerback. It is used to check the answerback.

APC. Application control.

API. Application programming interface.

APPC. Advanced Program-to-Program Communication based on SNA LU 6.2 protocols.

APPL. A VTAM definition statement used to define a VTAM application program.

application programming interface (API). An interface that programs can use to exchange data.

application support filter (ASF). In MERVA Link, a user-written program that can control and modify any data exchanged between the Application Support Layer and the Message Transfer Layer.

application support process (ASP). An executing instance of an application support program. Each application support process is associated with an ASP entry in the partner table. An ASP that handles outgoing messages is a *sending ASP*; one that handles incoming messages is a *receiving ASP*.

application support program (ASP). In MERVA Link, a program that exchanges messages and reports with a specific remote partener ASP. These two programs must agree on which conversation protocol they are to use.

ASCII. American Standard Code for Information Interchange. The standard code, using a coded set consisting of 7-bit coded characters (8 bits including parity check), used for information interchange among data processing systems, data communication systems, and associated equipment. The ASCII set consists of control characters and graphic characters.

ASF. Application support filter.

ASF. (1) Application support process. (2) Application support program.

ASPDU. Application support protocol data unit, which is defined in the MERVA Link P2 protocol.

authentication. The SWIFT security check used to ensure that a message has not changed during transmission, and that it was sent by an authorized sender.

authenticator key. A set of alphanumeric characters used for the authentication of a message sent via the SWIFT network.

authenticator-key file. The file that stores the keys used during the authentication of a message. The file contains a record for each of your financial institution's correspondents.

B

Back-to-Back (BTB). A MERVA Link function that enables ASPs to exchange messages in the local MERVA Link node without using data communication services.

bank identifier code. A 12-character code used to identify a bank within the SWIFT network. Also called a SWIFT address. The code consists of the following subcodes:
- The bank code (4 characters)
- The ISO country code (2 characters)
- The location code (2 characters)
- The address extension (1 character)

- The branch code (3 characters) for a SWIFT user institution, or the letters "BIC" for institutions that are not SWIFT users.

Basic Security Manager (BSM). A component of VSE/ESA Version 2.4 that is invoked by the System Authorization Facility, and used to ensure signon and transaction security.

BIC. Bank identifier code.

BIC Bankfile. A tape of bank identifier codes supplied by S.W.I.F.T.

BIC Database Plus Tape. A tape of financial institutions and currency codes, supplied by S.W.I.F.T. The information is compiled from various sources and includes national, international, and cross-border identifiers.

BIC Directory Update Tape. A tape of bank identifier codes and currency codes, supplied by S.W.I.F.T., with extended information as published in the printed BIC Directory.

body. The second part of an IM-ASPDU. It contains the actual application data or the message text that the IM-AMPDU transfers.

BSC. Binary synchronous control.

BSM. Basic Security Manager.

BTB. Back-to-back.

buffer. A storage area used by MERVA programs to store a message in its internal format. A buffer has an 8-byte prefix that indicates its length.

C

CBT. SWIFT computer-based terminal.

CCSID. Coded character set identifier.

CDS. Control data set.

central service. In MERVA, a service that uses resources that either require serialization of access, or are only available in the MERVA nucleus.

CF message. Confirmed message. When a sending MERVA Link system is informed of the successful delivery of a message to the receiving application, it routes the delivered application messages as CF messages, that is, messages of class CF, to an ACK wait queue or to a complete message queue.

COA. Confirm on arrival.

COD. Confirm on delivery.

coded character set identifier (CCSID). The name of a coded set of characters and their code point assignments.

commit. In MQSeries, to commit operations is to make the changes on MQSeries queues permanent. After putting one or more messages to a queue, a commit makes them visible to other programs. After getting one or more messages from a queue, a commit permanently deletes them from the queue.

confirm-on-arrival (COA) report. An MQSeries report message type created when a message is placed on that queue. It is created by the queue manager that owns the destination queue.

confirm-on-delivery (COD) report. An MQSeries report message type created when an application retrieves a message from the queue in a way that causes the message to be deleted from the queue. It is created by the queue manager.

control fields. In MERVA Link, fields that are part of a MERVA message on the queue data set and of the message in the TOF. Control fields are written to the TOF at nesting identifier 0. Messages in SWIFT format do not contain control fields.

correspondent. An institution to which your institution sends and from which it receives messages.

correspondent identifier. The 11-character identifier of the receiver of a telex message. Used as a key to retrieve information from the Telex correspondents file.

cross-system coupling facility. See *XCF*.

coupling services. In a sysplex, the functions of XCF that transfer data and status information among the members of a group that reside in one or more of the MVS systems in the sysplex.

couple data set. See *XCF couple data set*.

CTP. MERVA Link command transfer processor.

currency code file. A file containing the currency codes, together with the name, fraction length, country code, and country names.

D

daemon. A long-lived process that runs unattended to perform continuous or periodic systemwide functions.

DASD. Direct access storage device.

data area. An area of a predefined length and format on a panel in which data can be entered or displayed. A field can consist of one or more data areas.

data element. A unit of data that, in a certain context, is considered indivisible. In MERVA Link, a data element consists of a 2-byte data element length field, a 2-byte data-element identifier field, and a field of variable length containing the data element data.

datagram. In TCP/IP, the basic unit of information passed across the Internet environment. This type of message does not require a reply, and is the simplest type of message that MQSeries supports.

data terminal equipment. That part of a data station that serves as a data source, data link, or both, and provides for the data communication control function according to protocols.

DB2. A family of IBM licensed programs for relational database management.

dead-letter queue. A queue to which a queue manager or application sends messages that it cannot deliver. Also called *undelivered-message queue*.

dial-up number. A series of digits required to establish a connection with a remote correspondent via the public telex network.

direct service. In MERVA, a service that uses resources that are always available and that can be used by several requesters at the same time.

display mode. The mode (PROMPT or NOPROMPT) in which SWIFT messages are displayed. See *PROMPT mode* and *NOPROMPT mode.*

distributed queue management (DQM). In MQSeries message queuing, the setup and control of message channels to queue managers on other systems.

DQM. Distributed queue management.

DTE. Data terminal equipment.

E

EBCDIC. Extended Binary Coded Decimal Interchange Code. A coded character set consisting of 8-bit coded characters.

ECB. Event control block.

EDIFACT. Electronic Data Interchange for Administration, Commerce and Transport (a United Nations standard).

ESM. External security manager.

EUD. End-user driver.

exception report. An MQSeries report message type that is created by a message channel agent when a message is sent to another queue manager, but that message cannot be delivered to the specified destination queue.

external line format (ELF) messages. Messages that are not fully tokenized, but are stored in a single field in the TOF. Storing messages in ELF improves performance, because no mapping is needed, and checking is not performed.

external security manager (ESM). A security product that is invoked by the System Authorization Facility. RACF is an example of an ESM.

F

FDT. Field definition table.

field. In MERVA, a portion of a message used to enter or display a particular type of data in a predefined format. A field is located by its position in a message and by its tag. A field is made up of one or more data areas. See also *data area*.

field definition table (FDT). The field definition table describes the characteristics of a field; for example, its length and number of its data areas, and whether it is mandatory. If the characteristics of a field change depending on its use in a particular message, the definition of the field in the FDT can be overridden by the MCB specifications.

field group. One or several fields that are defined as being a group. Because a field can occur more than once in a message, field groups are used to distinguish them. A name can be assigned to the field group during message definition.

field group number. In the TOF, a number is assigned to each field group in a message in ascending order from 1 to 255. A particular field group can be accessed using its field group number.

field tag. A character string used by MERVA to identify a field in a network buffer. For example, for SWIFT field 30, the field tag is **:30:**.

FIN. Financial application.

FIN-Copy. The MERVA component used for SWIFT FIN-Copy support.

finite state machine. The theoretical base describing the rules of a service request’s state and the conditions to state transitions.

FMT/ESA. MERVA-to-MERVA Financial Message Transfer/ESA.

form. A partially-filled message containing data that can be copied for a new message of the same message type.

G

GPA. General purpose application.

H

HFS. Hierarchical file system.

hierarchical file system (HFS). A system for organizing files in a hierarchy, as in a UNIX system. OS/390 UNIX System Services files are organized in an HFS. All files are members of a directory, and each directory is in turn a member of a directory at a higher level in the HFS. The highest level in the hierarchy is the root directory.

I

IAM. Interapplication messaging (a MERVA Link message exchange protocol).

IM-ASPDU. Interapplication messaging application support protocol data unit. It contains an application message and consists of a heading and a body.

incore request queue. Another name for the request queue to emphasize that the request queue is held in memory instead of on a DASD.

InetD. Internet Daemon. It provides TCP/IP communication services in the OS/390 USS environment.

initiation queue. In MQSeries, a local queue on which the queue manager puts trigger messages.

input message. A message that is input into the SWIFT network. An input message has an input header.

INTERCOPE TelexBox. This telex box supports various national conventions for telex procedures and protocols.

interservice communication. In MERVA ESA, a facility that enables communication among services if MERVA ESA is running in a multisystem environment.

intertask communication. A facility that enables application programs to communicate with the MERVA nucleus and so request a central service.

IP. Internet Protocol.

IP message. In-process message. A message that is in the process of being transferred to another application.

ISC. Intersystem communication.

ISN. Input sequence number.

ISN acknowledgment. A collective term for the various kinds of acknowledgments sent by the SWIFT network.

ISO. International Organization for Standardization.

ITC. Intertask communication.

J

JCL. Job control language.

journal. A chronological list of records detailing MERVA actions.

journal key. A key used to identify a record in the journal.

journal service. A MERVA central service that maintains the journal.

K

KB. Kilobyte (1024 bytes).

key. A character or set of characters used to identify an item or group of items. For example, the user ID is the key to identify a user file record.

key-sequenced data set (KSDS). A VSAM data set whose records are loaded in key sequence and controlled by an index.

keyword parameter. A parameter that consists of a keyword, followed by one or more values.

KSDS. Key-sequenced data set.

L

LAK. Login acknowledgment message. This message informs you that you have successfully logged in to the SWIFT network.

large message. A message that is stored in the large message cluster (LMC). The maximum length of a message to be stored in the VSAM QDS is 31900 bytes. Messages up to 2MB can be stored in the LMC. For queue management using DB2 no distinction is made between messages and large messages.

large queue element. A queue element that is larger than the smaller of:

- The limiting value specified during the customization of MERVA
- 32KB

LC message. Last confirmed control message. It contains the message-sequence number of the application or acknowledgment message that was last confirmed; that is, for which the sending MERVA Link system most recently received confirmation of a successful delivery.

LDS. Logical data stream.

LMC. Large message cluster.

LNK. Login negative acknowledgment message. This message indicates that the login to the SWIFT network has failed.

local queue. In MQSeries, a queue that belongs to a local queue manager. A local queue can contain a list of messages waiting to be processed. Contrast with *remote queue*.

local queue manager. In MQSeries, the queue manager to which the program is connected, and that provides message queuing services to that program. Queue managers to which a program is not connected are remote queue managers, even if they are running on the same system as the program.

login. To start the connection to the SWIFT network.

LR message. Last received control message, which contains the message-sequence number of the application or acknowledgment message that was last received from the partner application.

LSN. Login sequence number.

LT. See *LTERM*.

LTC. Logical terminal control.

LTERM. Logical terminal. Logical terminal names have 4 characters in CICS and up to 8 characters in IMS.

LU. A VTAM logical unit.

M

maintain system history program (MSHP). A program used for automating and controlling various installation, tailoring, and service activities for a VSE system.

MCA. Message channel agent.

MCB. Message control block.

MERVA ESA. The IBM licensed program Message Entry and Routing with Interfaces to Various Applications for ESA.

MERVA Link. A MERVA component that can be used to interconnect several MERVA systems.

message. A string of fields in a predefined form used to provide or request information. See also *SWIFT financial message.*

message body. The part of the message that contains the message text.

message category. A group of messages that are logically related within an application.

message channel. In MQSeries distributed message queuing, a mechanism for moving messages from one queue manager to another. A message channel comprises two message channel agents (a sender and a receiver) and a communication link.

message channel agent (MCA). In MQSeries, a program that transmits prepared messages from a transmission queue to a communication link, or from a communication link to a destination queue.

message control block (MCB). The definition of a message, screen panel, net format, or printer layout made during customization of MERVA.

Message Format Service (MFS). A MERVA direct service that formats a message according to the medium to be used, and checks it for formal correctness.

message header. The leading part of a message that contains the sender and receiver of the message, the message priority, and the type of message.

Message Integrity Protocol (MIP). In MERVA Link, the protocol that controls the exchange of messages between partner ASPs. This protocol ensures that any loss of a message is detected and reported, and that no message is duplicated despite system failures at any point during the transfer process.

message-processing function. The various parts of MERVA used to handle a step in the message-processing route, together with any necessary equipment.

message queue. See *queue*.

Message Queue Interface (MQI). The programming interface provided by the MQSeries queue managers. It provides a set of calls that let application programs access message queuing services such as sending messages, receiving messages, and manipulating MQSeries objects.

Message Queue Manager (MQM). An IBM licensed program that provides message queuing services. It is part of the MQSeries set of products.

message reference number (MRN). A unique 16-digit number assigned to each message for identification purposes. The message reference number consists of an 8-digit domain identifier that is followed by an 8-digit sequence number.

message sequence number (MSN). A sequence number for messages transferred by MERVA Link.

message type (MT). A number, up to 7 digits long, that identifies a message. SWIFT messages are identified by a 3-digit number; for example SWIFT message type MT S100.

MFS. Message Format Service.

MIP. Message Integrity Protocol.

MPDU. Message protocol data unit, which is defined in P1.

MPP. In IMS, message-processing program.

MQA. MQ Attachment.

MQ Attachment (MQA). A MERVA feature that provides message transfer between MERVA and a user-written MQI application.

MQH. MQSeries queue handler.

MQI. Message queue interface.

MQM. Message queue manager.

MQS. MQSeries nucleus server.

MQSeries. A family of IBM licensed programs that provides message queuing services.

MQSeries nucleus server (MQS). A MERVA component that listens for messages on an MQI queue, receives them, extracts a service request, and passes it via the request queue handler to another MERVA ESA instance for processing.

MQSeries queue handler (MQH). A MERVA component that performs service calls to the Message Queue Manager via the provided Message Queue Interface.

MRN. Message reference number.

MSC. MERVA system control facility.

MSHP. Maintain system history program.

MSN. Message sequence number.

MT. Message type.

MTP. (1) Message transfer program. (2) Message transfer process.

MTS. Message Transfer System.

MTSP. Message Transfer Service Processor.

MTT. Message type table.

multisystem application. (1) An application program that has various functions distributed across MVS systems in a multisystem environment. (2) In XCF, an authorized application that uses XCF coupling services. (3) In MERVA ESA, multiple instances of MERVA ESA that are distributed among different MVS systems in a multisystem environment.

multisystem environment. An environment in which two or more MVS systems reside on one or more processors, and programs on one system can communicate with programs on the other systems. With XCF, the environment in which XCF services are available in a defined sysplex.

multisystem sysplex. A sysplex in which one or more MVS systems can be initialized as part of the sysplex. In a multisystem sysplex, XCF provides coupling services on all systems in the sysplex and requires an XCF couple data set that is shared by all systems. See also *single-system sysplex*.

MVS/ESA. Multiple Virtual Storage/Enterprise Systems Architecture.

N

namelist. An MQSeries for MVS/ESA object that contains a list of queue names.

nested message. A message that is composed of one or more message types.

nested message type. A message type that is contained in another message type. In some cases, only part of a message type (for example, only the mandatory fields) is nested, but this “partial” nested message type is also considered to be nested. For example, SWIFT MT 195 could be used to request information about a SWIFT MT 100 (customer transfer). The SWIFT MT 100 (or at least its mandatory fields) is then nested in SWIFT MT 195.

nesting identifier. An identifier (a number from 2 to 255) that is used to access a nested message type.

network identifier. A single character that is placed before a message type to indicate which network is to be used to send the message; for example, **S** for SWIFT

network service access point (NSAP). The endpoint of a network connection used by the SWIFT transport layer.

NOPROMPT mode. One of two ways to display a message panel. NOPROMPT mode is only intended for experienced SWIFT Link users who are familiar with the structure of SWIFT messages. With NOPROMPT mode, only the SWIFT header, trailer, and pre-filled fields and their tags are displayed. Contrast with *PROMPT mode*.

NSAP. Network service access point.

nucleus server. A MERVA component that processes a service request as selected by the request queue handler. The service a nucleus server provides and the way it provides it is defined in the nucleus server table (DSLNSVT).

O

object. In MQSeries, objects define the properties of queue managers, queues, process definitions, and namelists.

occurrence. See *repeatable sequence*.

option. One or more characters added to a SWIFT field number to distinguish among different layouts for and meanings of the same field. For example, SWIFT field 60 can have an option F to identify a first opening balance, or M for an intermediate opening balance.

origin identifier (origin ID). A 34-byte field of the MERVA user file record. It indicates, in a MERVA and SWIFT Link installation that is shared by several banks, to which of these banks the user belongs. This lets the user work for that bank only.

OSN. Output sequence number.

OSN acknowledgment. A collective term for the various kinds of acknowledgments sent to the SWIFT network.

output message. A message that has been received from the SWIFT network. An output message has an output header.

P

P1. In MERVA Link, a peer-to-peer protocol used by cooperating message transfer processes (MTPs).

P2. In MERVA Link, a peer-to-peer protocol used by cooperating application support processes (ASPs).

P3. In MERVA Link, a peer-to-peer protocol used by cooperating command transfer processors (CTPs).

packet switched public data network (PSPDN). A public data network established and operated by network common carriers or telecommunication administrations for providing packet-switched data transmission.

panel. A formatted display on a display terminal. Each page of a message is displayed on a separate panel.

parallel processing. The simultaneous processing of units of work by several servers. The units of work can be either transactions or subdivisions of larger units of work.

parallel sysplex. A sysplex that uses one or more coupling facilities.

partner table (PT). In MERVA Link, the table that defines how messages are processed. It consists of a header and different entries, such as entries to specify the message-processing parameters of an ASP or MTP.

PCT. Program Control Table (of CICS).

PDE. Possible duplicate emission.

PDU. Protocol data unit.

PF key. Program-function key.

positional parameter. A parameter that must appear in a specified location relative to other parameters.

PREMIUM. The MERVA component used for SWIFT PREMIUM support.

process definition object. An MQSeries object that contains the definition of an MQSeries application. A queue manager uses the definitions contained in a process definition object when it works with trigger messages.

program-function key. A key on a display terminal keyboard to which a function (for example, a command) can be assigned. This lets you execute the function (enter the command) with a single keystroke.

PROMPT mode. One of two ways to display a message panel. PROMPT mode is intended for SWIFT Link users who are unfamiliar with the structure of SWIFT messages. With PROMPT mode, all the fields and tags are displayed for the SWIFT message. Contrast with *NOPROMPT mode*.

protocol data unit (PDU). In MERVA Link a PDU consists of a structured sequence of implicit and explicit data elements:

- Implicit data elements contain other data elements.
- Explicit data elements cannot contain any other data elements.

PSN. Public switched network.

PSPDN. Packet switched public data network.

PSTN. Public switched telephone network.

PT. Partner table.

PTT. A national post and telecommunication authority (post, telegraph, telephone).

Q

QDS. Queue data set.

QSN. Queue sequence number.

queue. (1) In MERVA, a logical subdivision of the MERVA queue data set used to store the messages associated with a MERVA message-processing function. A queue has the same name as the message-processing function with which it is associated. (2) In MQSeries, an object onto which message queuing applications can put messages, and from which they can get messages. A queue is owned and maintained by a queue manager. See also *request queue*.

queue element. A message and its related control information stored in a data record in the MERVA ESA Queue Data Set.

queue management. A MERVA service function that handles the storing of messages in, and the retrieval of messages from, the queues of message-processing functions.

queue manager. (1) An MQSeries system program that provides queueing services to applications. It provides an application programming interface so that programs can access messages on the queues that the queue manager owns. See also *local queue manager* and *remote queue manager*. (2) The MQSeries object that defines the attributes of a particular queue manager.

queue sequence number (QSN). A sequence number that is assigned to the messages stored in a logical queue by MERVA ESA queue management in ascending order. The QSN is always unique in a queue. It is reset to zero when the queue data set is formatted, or when a queue management restart is carried out and the queue is empty.

R

RACF. Resource Access Control Facility.

RBA. Relative byte address.

RC message. Recovered message; that is, an IP message that was copied from the control queue of an inoperable or closed ASP via the **recover** command.

ready queue. A MERVA queue used by SWIFT Link to collect SWIFT messages that are ready for sending to the SWIFT network.

remote queue. In MQSeries, a queue that belongs to a remote queue manager. Programs can put messages on remote queues, but they cannot get messages from remote queues. Contrast with *local queue*.

remote queue manager. In MQSeries, a queue manager is remote to a program if it is not the queue manager to which the program is connected.

repeatable sequence. A field or a group of fields that is contained more than once in a message. For example, if the SWIFT fields 20, 32, and 72 form a sequence, and if this sequence can be repeated up to 10 times in a message, each sequence of the fields 20, 32, and 72 would be an occurrence of the repeatable sequence.

In the TOF, the occurrences of a repeatable sequence are numbered in ascending order from 1 to 32767 and can be referred to using the occurrence number.

A repeatable sequence in a message may itself contain another repeatable sequence. To identify an occurrence within such a nested repeatable sequence, more than one occurrence number is necessary.

reply message. In MQSeries, a type of message used for replies to request messages.

reply-to queue. In MQSeries, the name of a queue to which the program that issued an MQPUT call wants a reply message or report message sent.

report message. In MQSeries, a type of message that gives information about another message. A report message usually indicates that the original message cannot be processed for some reason.

request message. In MQSeries, a type of message used for requesting a reply from another program.

request queue. The queue in which a service request is stored. It resides in main storage and consists of a set of request queue elements that are chained in different queues:

- Requests waiting to be processed
- Requests currently being processed
- Requests for which processing has finished

request queue handler (RQH). A MERVA ESA component that handles the queueing and scheduling of service requests. It controls the request processing of a nucleus server according to rules defined in the finite state machine.

Resource Access Control Facility (RACF). An IBM licensed program that provides for access control by identifying and verifying users to the system, authorizing access to protected resources, logging detected unauthorized attempts to enter the system, and logging detected accesses to protected resources.

retype verification. See *verification*.

routing. In MERVA, the passing of messages from one stage in a predefined processing path to the next stage.

RP. Regional processor.

RQH. Request queue handler.

RRDS. Relative record data set.

S

SAF. System Authorization Facility.

SCS. SNA character string

SCP. System control process.

SDI. Sequential data set input. A batch utility used to import messages from a sequential data set or a tape into MERVA ESA queues.

SDO. Sequential data set output. A batch utility used to export messages from a MERVA ESA queue to a sequential data set or a tape.

SDY. Sequential data set system printer. A batch utility used to print messages from a MERVA ESA queue.

service request. A type of request that is created and passed to the request queue handler whenever a nucleus server requires a service that is not currently available.

sequence number. A number assigned to each message exchanged between two nodes. The number is increased by one for each successive message. It starts from zero each time a new session is established.

sign off. To end a session with MERVA.

sign on. To start a session with MERVA.

single-system sysplex. A sysplex in which only one MVS system can be initialized as part of the sysplex. In a single-system sysplex, XCF provides XCF services on the system, but does not provide signalling services between MVS systems. A single-system sysplex requires an XCF couple data set. See also *multisystem sysplex*.

small queue element. A queue element that is smaller than the smaller of:

- The limiting value specified during the customization of MERVA
- 32KB

SMP/E. System Modification Program Extended.

SN. Session number.

SNA. Systems network architecture.

SNA character string. In SNA, a character string composed of EBCDIC controls, optionally mixed with user data, that is carried within a request or response unit.

SPA. Scratch pad area.

SQL. Structured Query Language.

SR-ASPDU. The status report application support PDU, which is used by MERVA Link for acknowledgment messages.

SSN. Select sequence number.

subfield. A subdivision of a field with a specific meaning. For example, the SWIFT field 32 has the subfields date, currency code, and amount. A field can have several subfield layouts depending on the way the field is used in a particular message.

SVC. (1) Switched Virtual Circuit. (2) Supervisor call instruction.

S.W.I.F.T. (1) Society for Worldwide Interbank Financial Telecommunication s.c. (2) The network provided and managed by the Society for Worldwide Interbank Financial Telecommunication s.c.

SWIFT address. Synonym for *bank identifier code*.

SWIFT Correspondents File. The file containing the bank identifier code (BIC), together with the name, postal address, and zip code of each financial institution in the BIC Directory.

SWIFT financial message. A message in one of the SWIFT categories 1 to 9 that you can send or receive via the SWIFT network. See *SWIFT input message* and *SWIFT output message*.

SWIFT header. The leading part of a message that contains the sender and receiver of the message, the message priority, and the type of message.

SWIFT input message. A SWIFT message with an input header to be sent to the SWIFT network.

SWIFT link. The MERVA ESA component used to link to the SWIFT network.

SWIFT network. Refers to the SWIFT network of the Society for Worldwide Interbank Financial Telecommunication (S.W.I.F.T.).

SWIFT output message. A SWIFT message with an output header coming from the SWIFT network.

SWIFT system message. A SWIFT general purpose application (GPA) message or a financial application (FIN) message in SWIFT category 0.

switched virtual circuit (SVC). An X.25 circuit that is dynamically established when needed. It is the X.25 equivalent of a switched line.

sysplex. One or more MVS systems that communicate and cooperate via special multisystem hardware components and software services.

System Authorization Facility (SAF). An MVS or VSE facility through which MERVA ESA communicates with an external security manager such as RACF (for MVS) or the basic security manager (for VSE).

System Control Process (SCP). A MERVA Link component that handles the transfer of MERVA ESA commands to a partner MERVA ESA system, and the receipt of the command response. It is associated with a system control process entry in the partner table.

System Modification Program Extended (SMP/E). A licensed program used to install software and software changes on MVS systems.

Systems Network Architecture (SNA). The description of the logical structure, formats, protocols, and operating sequences for transmitting information units through, and for controlling the configuration and operation of, networks.

T

tag. A field identifier.

TCP/IP. Transmission Control Protocol/Internet Protocol.

Telex Correspondents File. A file that stores data about correspondents. When the user enters the corresponding nickname in a Telex message, the corresponding information in this file is automatically retrieved and entered into the Telex header area.

telex header area. The first part of the telex message. It contains control information for the telex network.

telex interface program (TXIP). A program that runs on a Telex front-end computer and provides a communication facility to connect MERVA ESA with the Telex network.

Telex Link. The MERVA ESA component used to link to the public telex network via a Telex substation.

Telex substation. A unit comprised of the following:

- Telex Interface Program
- A Telex front-end computer
- A Telex box

Terminal User Control Block (TUCB). A control block containing terminal-specific and user-specific information used for processing messages for display devices such as screen and printers.

test key. A key added to a telex message to ensure message integrity and authorized delivery. The test key is an integer value of up to 16 digits, calculated manually or by a test-key processing program using the significant information in the message, such as amounts, currency codes, and the message date.

test-key processing program. A program that automatically calculates and verifies a test key. The Telex Link supports panels for input of test-key-related data and an interface for a test-key processing program.

TFD. Terminal feature definitions table.

TID. Terminal identification. The first 9 characters of a bank identifier code (BIC).

TOF. Originally the abbreviation of *tokenized form*, the TOF is a storage area where messages are stored so that their fields can be accessed directly by their field names and other index information.

TP. Transaction program.

transaction. A specific set of input data that triggers the running of a specific process or job; for example, a message destined for an application program.

transaction code. In IMS and CICS, an alphanumeric code that calls an IMS message processing program or a CICS transaction. Transaction codes have 4 characters in CICS and up to 8 characters in IMS.

Transmission Control Protocol/Internet Protocol (TCP/IP). A set of communication protocols that support peer-to-peer connectivity functions for both local and wide area networks.

transmission queue. In MQSeries, a local queue on which prepared messages destined for a remote queue manager are temporarily stored.

trigger event. In MQSeries, an event (such as a message arriving on a queue) that causes a queue manager to create a trigger message on an initiation queue.

trigger message. In MQSeries, a message that contains information about the program that a trigger monitor is to start.

trigger monitor. In MQSeries, a continuously-running application that serves one or more initiation queues. When a trigger message arrives on an initiation queue, the trigger monitor retrieves the message. It uses the information in the trigger message to start a process that serves the queue on which a trigger event occurred.

triggering. In MQSeries, a facility that allows a queue manager to start an application automatically when predetermined conditions are satisfied.

TUCB. Terminal User Control Block.

TXIP. Telex interface program.

U

UMR. Unique message reference.

unique message reference (UMR). An optional feature of MERVA ESA that provides each message with a unique identifier the first time it is placed in a queue. It is composed of a MERVA ESA installation name, a sequence number, and a date and time stamp.

UNIT. A group of related literals or fields of an MCB definition, or both, enclosed by a DSLLUNIT and DSLLUEND macroinstruction.

UNIX System Services (USS). A component of OS/390, formerly called OpenEdition (OE), that creates a UNIX environment that conforms to the XPG4 UNIX 1995 specifications, and provides two open systems interfaces on the OS/390 operating system:

- An application program interface (API)
- An interactive shell interface

UN/EDIFACT. United Nations Standard for Electronic Data Interchange for Administration, Commerce and Transport.

USE. S.W.I.F.T. User Security Enhancements.

user file. A file containing information about all MERVA ESA users; for example, which functions each user is allowed to access. The user file is encrypted and can only be accessed by authorized persons.

user identification and verification. The acts of identifying and verifying a RACF-defined user to the system during logon or batch job processing. RACF identifies the user by the user ID and verifies the user by the password or operator identification card supplied during logon processing or the password supplied on a batch JOB statement.

USS. UNIX System Services.

V

verification. Checking to ensure that the contents of a message are correct. Two kinds of verification are:

- Visual verification: you read the message and confirm that you have done so
- Retype verification: you reenter the data to be verified

Virtual LU. An LU defined in MERVA Extended Connectivity for communication between MERVA and MERVA Extended Connectivity.

Virtual Storage Access Method (VSAM). An access method for direct or sequential processing of fixed and variable-length records on direct access devices. The records in a VSAM data set or file can be organized in logical sequence by a key field (key sequence), in the physical sequence in which they are written on the data set or file (entry sequence), or by relative-record number.

Virtual Telecommunications Access Method (VTAM). An IBM licensed program that controls communication and the flow of data in an SNA network. It provides single-domain, multiple-domain, and interconnected network capability.

VSAM. Virtual Storage Access Method.

VTAM. Virtual Telecommunications Access Method (IBM licensed program).

W

Windows NT service. A type of Windows NT application that can run in the background of the Windows NT operating system even when no user is logged on. Typically, such a service has no user interaction and writes its output messages to the Windows NT event log.

X

X.25. An ISO standard for interface to packet switched communications services.

XCF. Abbreviation for *cross-system coupling facility*, which is a special logical partition that provides high-speed caching, list processing, and locking functions in a sysplex. XCF provides the MVS coupling services that allow authorized programs on MVS systems in a multisystem environment to communicate with (send data to and receive data from) authorized programs on other MVS systems.

XCF couple data sets. A data set that is created through the XCF couple data set format utility and, depending on its designated type, is shared by some or all of the MVS systems in a sysplex. It is accessed only by XCF and contains XCF-related data about the sysplex, systems, applications, groups, and members.

XCF group. The set of related members defined to SCF by a multisystem application in which members of the group can communicate with (send data to and receive data from) other members of the same group. All MERVA systems working together in a sysplex must pertain to the same XCF group.

XCF member. A specific function of a multisystem application that is defined to XCF and assigned to a group by the multisystem application. A member resides on one system in a sysplex and can use XCF services to communicate with other members of the same group.

Bibliography

MERVA ESA Publications

- *MERVA for ESA Version 4: Application Programming Interface Guide*, SH12-6374
- *MERVA for ESA Version 4: Advanced MERVA Link*, SH12-6390
- *MERVA for ESA Version 4: Concepts and Components*, SH12-6381
- *MERVA for ESA Version 4: Customization Guide*, SH12-6380
- *MERVA for ESA Version 4: Diagnosis Guide*, SH12-6382
- *MERVA for ESA Version 4: Installation Guide*, SH12-6378
- *MERVA for ESA Version 4: Licensed Program Specifications*, GH12-6373
- *MERVA for ESA Version 4: Macro Reference*, SH12-6377
- *MERVA for ESA Version 4: Messages and Codes*, SH12-6379
- *MERVA for ESA Version 4: Operations Guide*, SH12-6375
- *MERVA for ESA Version 4: System Programming Guide*, SH12-6366
- *MERVA for ESA Version 4: User's Guide*, SH12-6376

MERVA ESA Components Publications

- *MERVA Automatic Message Import/Export Facility: User's Guide*, SH12-6389
- *MERVA Connection/NT*, SH12-6339
- *MERVA Connection/400*, SH12-6340
- *MERVA Directory Services*, SH12-6367
- *MERVA Extended Connectivity: Installation and User's Guide*, SH12-6157
- *MERVA Message Processing Client for Windows NT: User's Guide*, SH12-6341
- *MERVA-MQI Attachment User's Guide*, SH12-6714
- *MERVA Traffic Reconciliation*, SH12-6392
- *MERVA USE: Administration Guide*, SH12-6338
- *MERVA USE & Branch for Windows NT: User's Guide*, SH12-6334
- *MERVA USE & Branch for Windows NT: Installation and Customization Guide*, SH12-6335
- *MERVA USE & Branch for Windows NT: Application Programming Guide*, SH12-6336
- *MERVA USE & Branch for Windows NT: Diagnosis Guide*, SH12-6337
- *MERVA USE & Branch for Windows NT: Migration Guide*, SH12-6393
- *MERVA USE & Branch for Windows NT: Installation and Customization Guide*, SH12-6335
- *MERVA Workstation Based Functions*, SH12-6383

Other IBM Publications

- *DB2 Administration Guide*, S10J-8157
- *DB2 Building Applications for Windows and OS/2 Environment*, S10J-8160
- *DB2 API Reference*, S10J-8167
- *DB2 Troubleshooting Guide*, S10J-8169
- *eNetwork Personal Communications Version 4.2 for Windows 95 and Windows NT Quick Beginnings*, GC31-8476
- *eNetwork Personal Communications Version 4.2 for Windows 95 and Windows NT Reference*, GC31-8477
- *CID Enablement Guidelines*, S10H-9666
- *CICS-RACF Security Guide*, SC33-1185
- *ITSC Redbook APPC Security: MVS/ESA, CICS/ESA, and OS/2*, GG24-3960
- *IMS/ESA Version 4 Data Communication Administration Guide*, SC26-3060
- *MQSeries Application Programming Reference*, SC33-1673

S.W.I.F.T. Publications

The following are published by the Society for Worldwide Interbank Financial Telecommunication, s.c., in La Hulpe, Belgium:

- *S.W.I.F.T. User Handbook*
- *S.W.I.F.T. Dictionary*
- *S.W.I.F.T. FIN Security Guide*
- *S.W.I.F.T. Card Readers User Guide*

Index

Readers' Comments — We'd Like to Hear from You

MERVA ESA Components
MERVA USE Administration Guide
Version 4 Release 1

Publication No. SH12-6338-03

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	□	□	□	□	□

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	□	□	□	□	□
Complete	□	□	□	□	□
Easy to find	□	□	□	□	□
Easy to understand	□	□	□	□	□
Well organized	□	□	□	□	□
Applicable to your tasks	□	□	□	□	□

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? □ Yes □ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name

Address

Company or Organization

Phone No.

Readers' Comments — We'd Like to Hear from You

SH12-6338-03

Cut or Fold
Along Line

Fold and Tape **Please do not staple** Fold and Tape

PLACE
POSTAGE
STAMP
HERE

IBM Deutschland Entwicklung GmbH
Information Development & User Centered Design
Dept. 0446
Schoenaicher Strasse 220
71032 Boeblingen
Germany

Fold and Tape **Please do not staple** Fold and Tape

Cut or Fold
Along Line

Program Number: 5648-B30

SH12-6338-03