Installing the Standards Library Version 5.1

The Standards Library Version 5.1 installation is located on the Gentran Integration Suite Version 4.3 DVD. The following .jar files are used in the installation process:

.jar file	Location on the Gentran Integration Suite DVD	Function
translator_5100.jar	./standards directory	Installs the Standards Library Version 5.1 translator.
standards_5100.jar	./standards directory	Installs the Standards Library Version 5.1 standards.
swiftnet_5100.jar	./standards directory	Installs the Standards Library Version 5.1 SWIFTNet MEFG Server.

Complete the following instructions to install Standards Library Version 5.1:

Note: If you are installing the Standards Library Version 5.1 in a clustered environment, you need to perform the following steps on each node in the cluster.

1. Install Gentran Integration Suite Version 4.3 by following the *Gentran Integration Suite Installation* documentation.
2. Copy the following .jar files from the Gentran Integration Suite Version 4.3 DVD to your local file system:
   - translator_5100.jar
   - standards_5100.jar

Note: You need to install the **translator_5100.jar** file prior to installing the **standards_5100.jar** file. Both .jar files are necessary to complete the Standards Library Version 5.1 installation.

3. If you are using the SWIFTNet Server adapter in your current installation, prior to installing this new version of the Standards Library, you need to note the values you configured for the following parameters in the SWIFTNet Server adapter:
   - GIS HTTP Server Adapter Port
   - Config (for the RA1 instance)
   - Bin (for the RA1 instance)
   - Lib (for the RA1 instance)

- Category (for the RA1 instance)
- Config (for the RA2 instance, if used)
- Bin (for the RA2 instance, if used)
- Lib (for the RA2 instance, if used)
- Category (for the RA2 instance, if used)

These parameters may be overwritten during the upgrade process (each replaced with the default value for that parameter). If these parameters are overwritten, after the upgrade process is complete, you need to restore them to the original values you configured.

4. Change to the **<Gentran Integration Suite Installation Directory>/bin/** directory.
5. Install the **translator_5100.jar** file using the following commands:
   - If you are installing on a UNIX or iSeries operating system, type `./InstallService.sh full_path_to_directory_where_you_downloaded_jar_file/translator_5100.jar`
   - If you are installing on a Windows operating system, type `InstallService.cmd full_path_to_directory_where_you_downloaded_jar_file/translator_5100.jar`
6. Change to the **<Gentran Integration Suite Installation Directory>/bin/** directory.
7. Install the **standards_5100.jar** file using the following commands:
   - If you are installing on a UNIX or iSeries operating system, type `./InstallService.sh full_path_to_directory_where_you_downloaded_jar_file/standards_5100.jar`
   - If you are installing on a Windows operating system, type `InstallService.cmd full_path_to_directory_where_you_downloaded_jar_file/standards_5100.jar`
8. If you are not installing in a clustered environment, continue with step 8. If you are installing in a clustered environment, use the following commands:
   - If you are installing in a clustered environment on a UNIX or iSeries operating system, type `./startCluster.sh <node_number>`
   - If you are installing in a clustered environment on a Windows operating system, type `startCluster.cmd <node_number>`

   For example, if you are installing on Node 1 on a UNIX operating system, you would type `./startCluster.sh 1`
9. If you *do not* have a license for using SWIFTNet with Application, your installation is complete and you should disregard the remainder of this documentation.

   If you *do* have a license for using SWIFTNet with Application, complete the steps described in *Installing the Standards Library Version 5.1 SWIFTNet MEFG Server* on page 3.

Installing the Standards Library Version 5.1 SWIFTNet MEFG Server

Note: See *Using SWIFTNet* for complete information on implementing SWIFTNet with Application.

Overview

The Application SWIFTNet MEFG Server serves requests and receives messages to and from SWIFTNet, through a client application and a server application that communicate with the SWIFTNet network through the InterAct or FileAct protocol. The SWIFTNet MEFG Server operates independently from Application and includes all the APIs necessary to communicate with the SWIFTNet network.

Application enables you to use either InterAct or FileAct messaging with a store-and-forward option. The benefits of using store-and-forward include:

- The sender and receiver do not need to be online at the same time, as is required for real-time messaging.
- The sender is notified in the event delivery fails (and can optionally be notified upon delivery of the message).

Application also has a feature that provides you with failover support from real-time messages to store-and-forward (if there is a failure in real-time messaging, you can configure Application to automatically switch to store-and-forward messaging to increase your messaging success).

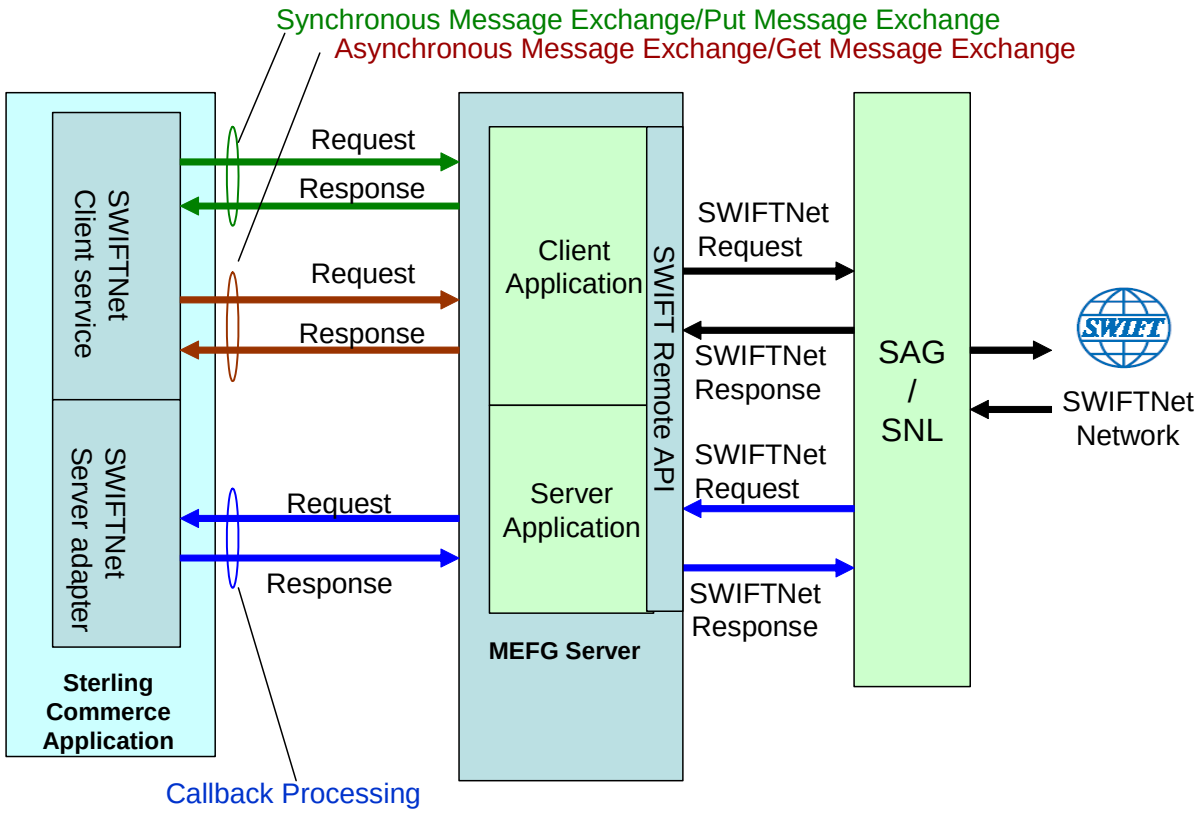
Note: This feature requires subscription to both real-time and store-and-forward services.

The SWIFTNet client application sends requests to the SWIFTNet network through the SWIFTNet Alliance Gateway/SWIFTNet Net Link (SAG/SNL) instance. The client application listens for requests from the Application SWIFTNet Client service, and interacts with SWIFTNet to obtain responses.

The SWIFTNet MEFG Server application receives requests from SWIFTNet. The server application listens for requests from SWIFTNet and interacts with Application to obtain responses. A request from the server application to Application calls the SWIFTNet Server adapter to process the request.

The SWIFTNet MEFG Server server application is started by enabling (and stopped by disabling) the Application SWIFTNet Server adapter. The starting and stopping of the server application is handled through the Command Line Adapter 2, which is built into the SWIFTNet Server adapter.

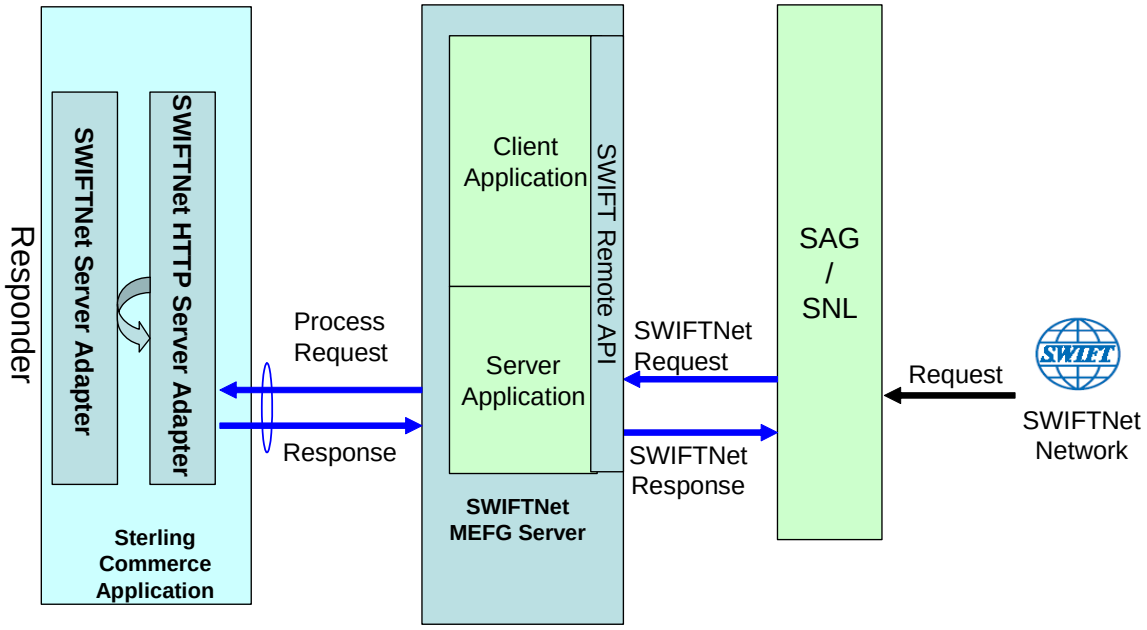
This diagram illustrates the process flow between Application and the SWIFTNet network through the SWIFTNet MEFG Server:



The administration of the SWIFTNet MEFG Server is through SWIFTNet Server adapter, including enabling and disabling the SWIFTNet MEFG Server.

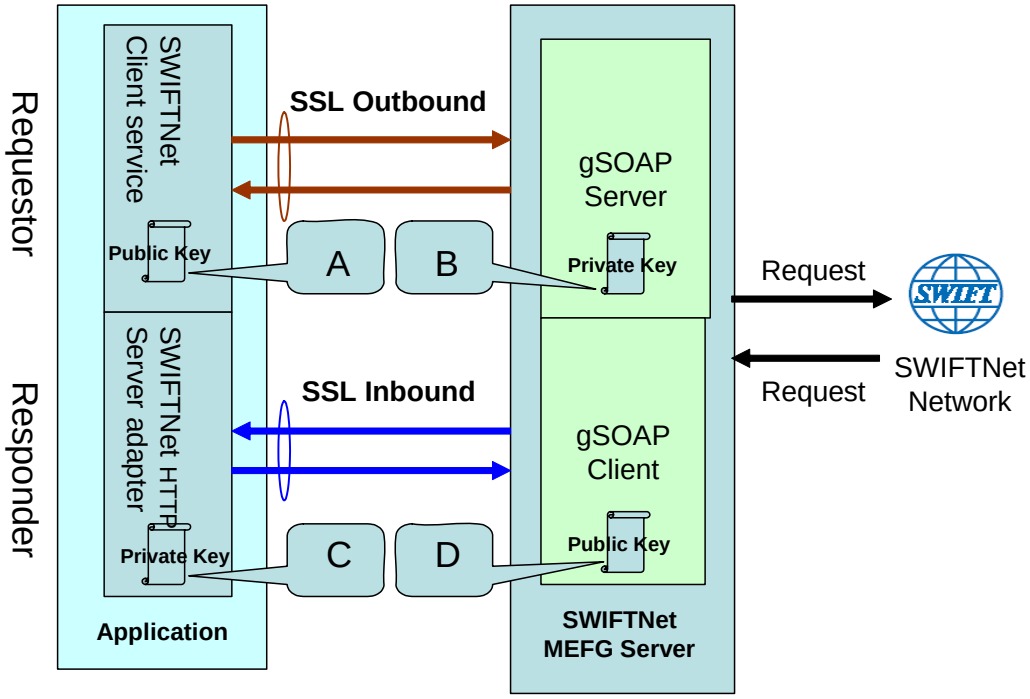
If you use the SWIFTNet HTTP Server adapter in conjunction with the SWIFTNet Server adapter to use Secure Sockets Layer (SSL), the SWIFTNet HTTP Server adapter accepts the forwarded request from the SWIFTNet MEFG Server and provides secure authentication.

This diagram illustrates the process flow between Application and the SWIFTNet network through the SWIFTNet MEFG Server (using the SWIFTNet HTTP Server adapter for SSL):



The SWIFTNet Client adapter (in conjunction with the SWIFTNet HTTP Server adapter) enables you to use Secure Sockets Layer (SSL) to provide secure authentication, using the SWIFTNet HTTP Server adapter to accept the forwarded request from the SWIFTNet MEFG Server. When you use SSL with Application, two channels are secured: an Outbound channel (Application acting as the Requestor) and an Inbound channel (Application acting as the Responder).

This diagram illustrates the process flow between Application and the SWIFTNet network through the SWIFTNet MEFG Server, including the Outbound and Inbound channels (using the SWIFTNet HTTP Server adapter for SSL):



You will need 2 pairs of certificates. The first pair belongs to the SWIFTNet MEFG Server (A and B in the diagram above) and is used to secure the outbound channel. The second pair of certificates belongs to Application (C and D in the diagram above) and is used to secure the inbound channel. In the above diagram, the callouts signify the following:

- A — A public key certificate file belongs to the SWIFTNet MEFG Server that is configured on the SWIFTNet Client service (the certificate is specified for the CA Certificate parameter).
- B — A private key certificate file that is stored on the SWIFTNet MEFG Server as a key file (which you configure through the SSL Configuration utility named sslUtil.jar in the SWIFTNet MEFG Server installation bin sub-directory).
- C — A private key certificate file that is configured on the SWIFTNet HTTP Server adapter (the certificate is specified for the System Cert parameter).
- D — A public key file that belongs to Application and is stored for the SWIFTNet MEFG Server as a CA Cert file or trusted list (that you configure through the SSL Configuration utility named sslUtil.jar in the SWIFTNet MEFG Server installation bin sub-directory).

Supported Platforms

The following platforms are supported for the SWIFTNet MEFG Server:

- Windows Server 2003 (Standard or Enterprise Edition) with Service Pack 1
- AIX 5.3 ML02

- SunOS 5.10

Client Application

The client application can exchange messages in synchronous or asynchronous mode for InterAct processing or can exchange messages in Put or Get mode for FileAct processing.

Synchronous Message Exchange

When the client application is communicating in synchronous mode messaging, the SWIFTNet Client service prepares the request and sends it to the SWIFTNet MEFG Server. Then, the client application on the SWIFTNet MEFG Server processes the request, performs the necessary communication exchange with the SWIFTNet SAG/SNL instance, and sends the request to the SWIFTNet network.

In synchronous mode, the client application is blocked until a response is received from the responder through the SAG/SNL instance. Once a response is received, it is sent to Application by the client application, and the response is placed in the primary document.

Asynchronous Message Exchange

In asynchronous mode, the SWIFTNet Client service prepares the request and sends it to the SWIFTNet MEFG Server. The client application on the SWIFTNet MEFG Server processes the request, performs the necessary communication exchange with the SWIFTNet SAG/SNL instance, and sends the request to the SWIFTNet network.

In asynchronous mode, the client application receives a response handle from the SAG/SNL instance. Using this response handle, the client application periodically checks with the SWIFTNet network to determine if a response is available . Once a response is received by Application, it is placed in the primary document.

Configuring the Client Application

There is no configuration necessary for the client application (either in synchronous or asynchronous mode or in Put or Get mode), but you must appropriately configure the SWIFTNet Client service to use the client application. See *SWIFTNet Client Service*.

Server Application

When a request from the SWIFTNet network arrives, the SWIFTNet SAG/SNL sends it to the SWIFTNet MEFG Server server application. The server application processes the request and forwards the request to Application. When Application receives the request, it invokes the SWIFTNet Server adapter to process the request. If store-and-forward messaging is in use, the message payload is placed in a mailbox.

Configuring the Server Application

There is no configuration necessary for the server application, but you must appropriately configure the SWIFTNet Server adapter to use the server application. See *SWIFTNet Server Adapter*.

SWIFTNet MEFG Server Installation Overview

The SWIFTNet MEFG Server installation consists of a sequence of related tasks. This table outlines the process flow you must follow to install the SWIFTNet MEFG Server:

Task Number	Description	For more information
1	Configure SAG/SNL.	*Configuring SAG/SNL* on page 9
2	Install and configure the SWIFTNet remote API.	*Installing the SWIFTNet Remote API (RA)* on page 11
3	Install OpenSSL software on the SWIFTNet MEFG Server host. **Note:** Optional—this is only required if you wish to enable SSL support between Application and SWIFTNet MEFG Server.	Obtain the OpenSSL installation software (Win32 OpenSSL v0.9.8e Light version) from: http://www.slproweb.com/products/Win32OpenSSL.html. On Solaris 10 and AIX 5.3 machines, the OpenSSL libraries should be included with your operating system, although they may be optionally installed. Please check the appropriate documentation for your operating system to ensure the OpenSSL software is installed.
4	Install the SWIFTNet MEFG Server.	*Installing the SWIFTNet MEFG Server* on page 12
5	Configure SSL between Application and the SWIFTNet MEFG Server. **Note:** Optional—this is only required if you wish to enable SSL support between Application and SWIFTNet MEFG Server.	*Configuring SSL Between Application and the SWIFTNet MEFG Server* on page 14
6	Install the Command Line Adapter 2 Client,	*Starting the Command Line Adapter 2 Client* on page 17
7	Configure and enable the SWIFTNet Server Adapter and start the SWIFTNet MEFG Server. **Note:** To monitor the status of the SWIFTNet MEFG Server, you need to select **Show Advanced Status** when you configure the SWIFTNet Server adapter.	*SWIFTNet Server Adapter*
8	Configure and enable the SWIFTNet Client Service.	*SWIFTNet Client Service*
9	Configure the SWIFTNet routing rule.	*SWIFTNet Routing Rule*
10	Configure Mailboxes (only if you are executing store-and-forward).	*Using Mailboxes*
11	Configure the SWIFTNetClient business process.	*SWIFTNetClient Business Process*

The following prerequisites must be met for the SWIFTNet MEFG Server to operate:

- The Command Line Adapter 2 client (CLA2Client) must be running to receive commands from Application to start and stop the SWIFTNet MEFG Server.
- The Command Line Adapter 2 client (CLA2Client) must be deployed on the same machine as the SWIFTNet MEFG Server.
- SWIFTNet Remote API (RA) must be installed on the same machine as the SWIFTNet MEFG Server.
- The SAG/SNL must be installed and configured with appropriate message partners and endpoints. See *Configuring SAG/SNL* on page 9.
- You must have a SWIFTNet Subscription for the InterAct and/or FileAct protocols.
- You use the same account to install the SWIFTNet MEFG Server as you used for the SAG/SNL installation.
- You have Java JDK 1.5 installed.
- You must install the SWIFTNet MEFG Server on either the Sun Solaris 5.10 operating system, Windows Server 2003 (Standard or Enterprise Edition) operating system, or AIX 5.3 operating system.

Configuring SAG/SNL

Complete the following steps to configure SAG/SNL for use with the SWIFTNet MEFG Server:

1. Log in as an administrator to the SWIFTAlliance Workstation.
2. Go to **Gateway Admin - Application Interface** and create the client and server message partners.

   Note: The client (type = Client) and server (type = Server) message partner names must match the names in the SWIFTNet MEFG Server configuration (<**SagMessagePartnerClientName**> and <**SagMessagePartnerServerName**>).
3. In the Application Interface module, for the server message partner, configure the parameters as follows:

Parameter	Configuration
Name	Name from the SWIFTNet MEFG Server configuration (<SagMessagePartnerServerName>).
Type	Server
Status	Enabled
Unit	None
Host Adapter	Remote API Host Adapter
Supported Message Formats	Select **Strict SNL Format**.
Additional Processing	Select **Remote API Host Adapter**.

4. In the Application Interface module, for the client message partner, configure the parameters as follows:

   Note: The Application Interface must be started.

Parameter	Configuration
Name	Name from the SWIFTNet MEFG Server configuration (<SagMessagePartnerClientName>).
Type	Client
Status	Enabled
Unit	None
Default Message Format for Emission (from Message Partner)	Strict SNL Format **Note:** Strict SNL Format is required by the API.
Supported Message Formats	Select **Strict SNL Format**. **Note:** Strict SNL Format is required by the API.
Additional Processing	**Note:** Do not select any additional processing options.

5. In the Endpoints module, for the server message partner, configure the endpoint parameters as follows to define where to route the messages:

Parameter	Configuration
Name	Name from the SWIFTNet MEFG Server configuration.
Destination	Application Interface:<Name from the SWIFTNet MEFG Server configuration>
Status	Enabled

6. In the Endpoints module, for the server message partner, configure the routing detail parameters as follows:

Parameter	Configuration
From	SWIFTNet Interface
Sequence	**Note:** This is the sequence number.
Name	Name from the SWIFTNet MEFG Server configuration (<SNLEndPoint>).
Status	Enabled
SNL Endpoint	None
Service Name	None
Request Type	None
Requestor DN	Relation: **Equals (=**) Parameter: **o=administrator,o=swift**

Parameter	Configuration
Responder DN	Relation: **Equals (=**) Parameter: **o=administrator,o=swift**
Traffic Type	None
Delivery Mode	None
Priority	None

7. In the Endpoints module, for the server message partner, configure the destination detail parameters as follows:

Parameter	Configuration
Interface	Application Interface
Application	Name of the SWIFTNet MEFG Server server application (<SagMessagePartnerServerName>).
Mode	Strict

Installing the SWIFTNet Remote API (RA)

You need to install the SWIFTNet Remote API on the machine on which the SWIFTNet MEFG Server will be installed. This is the software distributed by SWIFT, the API that the SWIFTNet MEFG Server uses to connect to the SWIFTNet SAG/SNL instance to link into a SAG.

Complete the following steps to install the RA:

1. Install the remote API on the machine where you will install SWIFTNet MEFG Server.
2. Configure the RA to point to the SAG instance you will be accessing, or if the RA is already configured, verify that it points to the correct SAG instance.
3. If you are installing on the Windows operating system, add **SYSTEM** to the Security settings, allow SYSTEM **Full Control** and select **Allow inheritable permissions from parent to propagate to this object**. See the documentation for the SWIFTNet Remote API for more information.
    a. Right-click and select **Properties** on <SWIFT RA API installdir>.
    b. Select the **Security** tab.
    c. Click **Add** to and select **SYSTEM**.
    d. Allow **Full Control** to **SYSTEM**.
    e. Select the **Allow inheritable permissions from parent to propagate this object** option.
    f. To confirm, navigate to the <Applicationinstalldir\SWIFTAlliance\RA\lib> directory, and right-click and select **Properties**.
    g. Select the **Security** tab.
    h. Verify that SYSTEM has Full Control permissions for the <Applicationinstalldir\SWIFTAlliance\RA\lib> directory.

4. If you are installing on a Windows operating system, following the SWIFT RA installation, you will need to set the PATH system environment variables:

   ```
   PATH : append <Swift RA API installdir>\bin;<Swift RA API installdir>\lib
   SWNET_HOME : <Swift RA API installdir>
   SWNET_CFG_PATH : <Swift RA API installdir>\Ra1\cfg;
   ```

Installing the SWIFTNet MEFG Server

Complete the following steps to install the Standards Library Version 5.1 SWIFTNet MEFG Server on a Solaris or AIX system:

Note: The installation script and the binary install are located on the Application installation DVD.

1. After installing Application, log on to your UNIX system using the same account as the one you used to install and configure SWIFTNet RA.
2. Download the **swiftnet_5100.jar** file from the installation DVD to the server where the SWIFTNet MEFG Server will be installed (the server on which SWIFTNet RA libraries are installed).
3. Type the following command to invoke the installation script and press **Enter**:

   ```
   [path to java bin]/java -jar swiftnet_5100.jar
   ```

4. Type the destination directory where you want to install the SWIFTNet MEFG Server and press **Enter**.
5. When you are prompted for confirmation, type **y** and press **Enter**.

   Note: To change the destination directory, type **n** and press **Enter**, and repeat step 4. If the destination directory does not have enough free disk space, the script suggests you delete enough files to provide the necessary disk space and then exit the installation.

   The installation script copies files from the distribution media to the destination directory and verifies that the correct number of files and blocks are copied.

   If you are installing on a UNIX operating system, you are notified that the installation is complete with the following message: Installation of Application SWIFTNet MEFG Server component is finished. You will need to configure this application in your Application user interface.

   If you are installing on a Windows operating system, continue with step 6.

6. If you are installing on the Windows operating system, you are required to enter the account user name in the format **DomainName\Username**. If it is a local user, type **.\Administrator**.
7. Type **Yes** to confirm the account user name and press **Enter**.
8. Type the correct account password, confirm the password, and press **Enter**.

   This installs the following service instances (you can verify this by checking **Control Panel** > **Administrative tools** > **Services**):

   - MEFGCommServer Service
   - MEFGSwiftnetServer Service Instance 1
   - MEFGSwiftnetServer Service Instance 2
   - MEFGCommSSLServer Service
   - MEFGSwiftnetSSLServer Service Instance 1

- MEFGSwiftnetSSLServer Service Instance 2

Note: If you have already installed any of these services, you will be notified through an error message containing Error Code 1073 that the service or services are already installed. If this occurs, you can unregister the services as follows:

a. Go to <MEFG Server directory>\bin.

b. Type the following and press **Enter** after each line:

To uninstall the MEFG Comm Server Service, type `MEFGCommServer.exe -u`

To uninstall the MEFG SWIFTNet Server Service Instance 1, type `MEFGSwiftnetServer.exe -u s1`

To uninstall the MEFG SWIFTNet Server Service Instance 2, type `MEFGSwiftnetServer.exe -u s2`

To uninstall the MEFG Comm SSL Server Service, type `MEFGCommSSLServer.exe -u`

To uninstall the MEFG SWIFTNet SSL Server Service Instance 1, type `MEFGSWIFTNetSSLServer.exe -u s1`

To uninstall the MEFG SWIFTNet SSL Server Service Instance 2, type `MEFGSWIFTNetSSLServer.exe -u s2`

9. If you are installing on a Windows operating system, you must add the following under the System variables (add the <MEFG installdir>\bin directory to the PATH environment variable just before the <SWIFT RA API installdir>\bin entry):

- PATH : append <MEFG installdir>\bin;

Caution: Insert this PATH variable *before* the SWIFT RA API installdir entries you added in step 4 of *Installing the SWIFTNet Remote API (RA)* on page 11 (the PATH variable must appear ahead of the library references in the System variable list).

- Allow the defined user to start MEFGSwiftnetServer and MEFGCommServer through the following steps:

a. Select **Control Panel** > **Administrative Tools** > **Local Security Settings**.

b. Select **Local Policies** > **User Rights Assignment**.

c. Double-click **Log on as a service** and assign the account user (that you entered during the installation process) to this setting.

10. If you are installing on an AIX operating system, you must modify the startup script **MEFGCommServer.sh** in the **<Swiftnet Server install>/bin** directory by passing option **-o 3**.

In the **MEFGCommServer.sh** startup script, locate this line:

```
${SWV2DIR}/bin/${DAEM_NAME} ${PORT} -s 0 -a ${SWV2DIR} 1>/dev/null
2>/dev/null &
```

And change it by adding the following (in boldface type):

```
${SWV2DIR}/bin/${DAEM_NAME} ${PORT} -o 3 -s 0 -a ${SWV2DIR} 1>/dev/null
2>/dev/null &
```

Configuring SSL Between Application and the SWIFTNet MEFG Server

Note: Optional—this is only required if you wish to enable SSL support between Application and SWIFTNet MEFG Server.

To configure SSL for the SWIFTNet MEFG Server, you must complete the following tasks:

Task Number	Description
1	Prepare the SSL certificates for the SWIFTNet MEFG Server.
2	Prepare the SSL certificates for Application.
3	Configure the SWIFTNet Server adapter.
4	Configure the SWIFTNetClient business process.
5	Configuring the SSL Setup on the SWIFTNet MEFG Server

Preparing the SSL Certificates for the SWIFTNet MEFG Server

To prepare the SSL certificates for use with the SWIFTNet MEFG Server, complete the following:

1. Create the keyfile that contains the private key for MEFGCommServer.

   You can use OpenSSL to generate the keyfile and certificate request file. Then, you can use this certificate request to ask the CA to generate the certificate and sign it for you.

Note: Note: When you use OpenSSL to generate the keyfile, you will be prompted to type in the password to protect the keyfile.

Note: Please take note of this password for later use. (step 8 of *Configuring the SSL Setup on the SWIFTNet MEFG Server*).

2. Import the signed certificates into the Application CA repository.
3. Note the CA Certificate ID and CA Certificate Name because you will need to use them in the SWIFTNetClient business process.
4. Create the CA Certificate that contains the cert (public key) for the Application side of this configuration.

Note: You can only do this after you have completed step 3 of *Preparing the SSL Certificates for* Application.

Preparing the SSL Certificates for Application

To prepare the SSL certificates for use with Application, complete the following:

1. Create a self-signed certificate on Application for the System certificate, including the following:
   - Select **Set Certificate Signing Bit**.
   - The name of this certificate must be the name of your server/domain so that Open SSL can properly validate the certificate.

Note: Alternatively, you can generate a certificate signing request using the Application Certificate Wizard, and ask a CA to sign your certificate. If you choose this option, include the following:

- Ensure your common name for the certificate matches the correct server name/domain name.
- Check in the key and the certificate (after the certificate is signed and returned by CA) to Application System Certificate.

2. When you create and check in the certificate, note the certificate name. This name will be used when you configure the SWIFTNet HTTP Server adapter (step 4 below).
3. Export the public key of the certificate you generated above. This public key will be used by the SWIFTNet MEFG Server (that you created in *Preparing the SSL Certificates for the SWIFTNet MEFG Server* on page 14) as CA certificate file (trusted list).
4. Configure the SWIFTNet HTTP Server adapter to use SSL and choose the System certificate you generated in step 1 of this procedure.

Note: Note the port number of the SWIFTNet HTTP Server adapter because this number must match the port you configure for the SWIFTNet Server adapter (below).

Configuring the SWIFTNet Server Adapter for SSL

To configure the SWIFTNet Server adapter for SSL, complete the following:

1. Select **Deployment > Services > Configuration**.
2. Search for SWIFTNet Server adapter or select it from the list and click **Go!**.
3. Click **Edit**.
4. For **GIS HTTP Server Adapter Port**, use the SSL port configured for the SWIFTNet HTTP Server adapter (above).
5. For **GIS Server IP**, type your exact server name/domain name. This must match with the server/domain name and also match the system certificate name you created in *Preparing the SSL Certificates for Application* on page 14.
6. Ensure that **Use SSL** is set to **True**.

Configuring the SWIFTNet HTTP Server Adapter for SSL

To configure the SWIFTNet HTTP Server adapter for SSL, complete the following:

1. Select **Deployment > Services > Configuration**.
2. Search for SWIFTNet HTTP Server adapter or select it from the list and click **Go!**.
3. Click **Edit**.
4. Ensure that **Use SSL** is set to **Must**.
5. For **System Certificate**, select the appropriate system certificate.

Configuring the SWIFTNet Client Service or Business Process for SSL

You must either configure the SWIFT Client service through the Application user interface or through the business process you create for the service.

To configure the SWIFTNet Client service for SSL, complete the following:

1. Select **Deployment > Services > Configuration**.

2. Search for SWIFTNet Client service or select it from the list and click **Go!**.
3. Click **Edit**.
4. Ensure that **Use SSL** is set to **Must**.
5. For **CA Certificate**, select the appropriate CA certificate. This is the certificate you imported in step 2 of *Preparing the SSL Certificates for the SWIFTNet MEFG Server* on page 14.

Alternatively, to configure the SWIFTNetClient business process, add the following to the BPML to ensure the SSL configuration for the SWIFTNet HTTP Client adapter is included:

Note: The **bold** lines indicate information that you need to modify to match your installation.

```
<<operation>
      <participant name="SWIFTNetClientService"/>
      <output message="handleClientRequest">
        <assign to="." from="*"></assign>
        <assign to="interfaceMode">interact</assign>
        <assign to="swiftOp">sync</assign>
        <assign to="requestorDN">o=ptscfrnn,o=swift</assign>
        <assign to="responderDN">o=ptscfrnn,o=swift</assign>
        <assign to="serviceName">swift.generic.ia!x</assign>
        <assign to="SnF">FALSE</assign>
        <assign to="nonRepudiation">FALSE</assign>
        <assign to="possibleDuplicate">FALSE</assign>
        <assign to="deliveryNotification">FALSE</assign>
        <assign to="UseSSL">TRUE</assign>
        <assign to="CipherStrength">All</assign>
  <assign to="CACertId">000.00.000.00:00000:10f3202f455:4337</assign>
      </output>
      <input message="testing">
        <assign to="." from="*"></assign>
      </input>
    </operation>
```

Note: The **CACertId** must match the CA Cert ID you configured for the SWIFTNet MEFG Server (step 2 of *Preparing the SSL Certificates for the SWIFTNet MEFG Server* on page 14).

Configuring the SSL Setup on the SWIFTNet MEFG Server

Prior to completing the next steps, ensure that you have completed the tasks in *Configuring SSL Between Application and the SWIFTNet MEFG Server* on page 14.

1. Type the following command to change to the directory where the SWIFTNet MEFG Server SSL Utilityjar is located:

   ```
   cd <SWIFTNet MEFG Server installdir>/bin
   ```

2. Type the following command to invoke the configuration script:

   ```
   dir *.jar
   ```

   You will see the sslUtil.jar file.

3. Type the following command:

   ```
   java -jar sslUtil.jar
   ```

4. Type the full path for the SWIFTNet MEFG Server home directory (the directory in which you just installed the SWIFTNet MEFG Server) and press **Enter**. You are prompted to confirm the directory.
5. Type **Yes** and press **Enter** to confirm. The configuration script verifies the structure of the directory to ensure that it is the valid installation directory for the SWIFTNet MEFG Server. If the directory is not valid (for example, the bin directory is missing from the path), you are prompted to retype the valid directory. Once a valid directory is entered, you proceed with the SSL configuration.
6. Type the full path to the private key location and press **Enter**. This key file belongs to the SWIFTNet MEFG Server and will used during the SSL "handshake."

Note: The key file contains your private key.

7. Type **Yes** and press **Enter** to confirm the path.
8. Type the correct password to access the key file and press **Enter**.
9. Type the password again to confirm it and press **Enter**.
10. Type the full path to the CA Certificate location and press **Enter**. This CA Certificate file contains the trusted certificates that are used during the SSL "handshake."
11. Type **Yes** and press **Enter** to confirm the path. The configuration completes and displays a message that the SSL configuration process has finished.

Configuring Fail-over Processing Using the SWIFTNet MEFG Server

To set up the SWIFTNet MEFG Server in a dual-active SAG configuration for fail-over processing, specify the following application interface definitions for the SWIFTNet Server adapter:

- ✦ active-active Configuration
- ✦ RA1 definitions for primary SAG (s1)
- ✦ RA2 definitions for alternate SAG (s2)

Note: Certificates and profiles must be available on the SAG where they are used. For fail-over processing, Puts and Gets try to connect to the first SAG specified for s1 and s2. If the connection fails, the Put and Get try to connect to the next SAG. If this connection also fails, the cycle is repeated if retry has been enabled.

Starting the Command Line Adapter 2 Client

The Command Line Adapter 2 client (CLA2Client) must be installed and run on a remote server. Complete the following steps to start the remote adapter implementation version of the Command Line Adapter 2:

1. Locate the client jar (CLA2Client.jar) in your Application installation that contains the necessary classes.
2. Move the client jar to the machine that will be running the remote Command Line Adapter 2 client.

   Note: This is the machine on which the SWIFTNet MEFG Server is installed.

3. Start the remote adapter implementation using the following command:

   ```
   [path to java bin]/java -jar [path to CLA2 Client jar file]/CLA2Client.jar
   <port> [debug]
   ```

   Note: The port (above) will be used when you configure the SWIFTNet Server adapter.

   This is an example of the command to start the remote adapter implementation:

   ```
   jdk1.5.0_11/bin/java -jar CLA2Client.jar 15699 debug
   ```

   Note: The [debug] option is not required, but you may find it helpful. When you upgrade Application, you will also need to obtain the corresponding new CLA2Client.jar file to avoid receiving a ClassConflict error.

Monitoring the Status of the SWIFTNet MEFG Server

To monitor the status of the SWIFTNet MEFG Server, you need to select **Show Advanced Status** when you configure the SWIFTNet Server adapter:

1. Select **Deployment > Services > Configuration**.
2. Search for SWIFTNet Server adapter or select it from the list and click **Go!**.

   Note: When you select the SWIFTNet Server adapter, make sure you also select the **Show Advanced Status** check box prior to clicking **Go!**. This enables you to view the Advanced Status column on the Services Configuration page to see whether the SWIFTNet MEFG Server is stopped or started.
3. Click **Edit**.
4. Specify field settings in the Admin Console. See *SWIFTNet Server Adapter.*
5. On the Confirm page, verify that the **Enable Service for Business Processes** check box is selected. This enables the adapter instance.

Starting and Stopping the SWIFTNet MEFG Server

To start and stop the SWIFTNet MEFG Server:

1. Select **Deployment > Services > Configuration**.
2. Search for SWIFTNet Server adapter or select it from the list and click **Go!**.

   Note: When you select the SWIFTNet Server adapter, make sure you also select the **Show Advanced Status** check box prior to clicking **Go!**. This enables you to view the Advanced Status column on the Services Configuration page to see whether the SWIFTNet MEFG Server is stopped or started.
3. Once the SWIFTNet Client adapter is configured and saved, click the **Enabled** check box on the Services Configuration page. This starts the SWIFTNet MEFG Server.

   Note: To stop the SWIFTNet MEFG Server, clear the **Enabled** check box on the Services Configuration page