Infor Enterprise Server eMessage Administration Guide

Release 10.7.x

Publication Information

Release: Infor LN 10.7.x
Publication Date: January 21, 2019
Document code: ln_10.7.x_lnttemessageag__en-us

Contents

About this guide

This document forms the system administration guide for the LN eMessage Connector. The contents include information on message handling, how LN uses the repository, the message log, addressing and printing reports.

Intended audience

This guide is intended for system administrators.

Related documents

You can find the documents in the product documentation section of the Infor Support Portal, as described in "Contacting Infor".

Contacting Infor

If you have questions about Infor products, go to Infor Concierge at https://concierge.infor.com/ and create a support incident.

If we update this document after the product release, we will post the new version on the Infor Support Portal. To access documentation, select **Search > Browse Documentation**. We recommend that you check this portal periodically for updated documentation.

If you have comments about Infor documentation, contact documentation@infor.com.

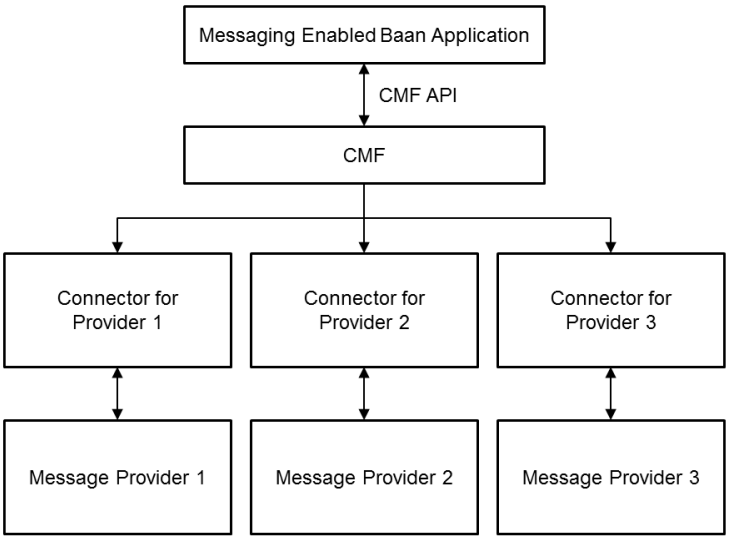
Chapter 1: Introduction

With the eMessage Connector you can build several classes of messaging applications within the context of an LN application. LN supports email messages. With the eMessage Connector, the LN application can send messages to a server-side service provider (SMTP).

The eMessage Connector and all the functions in its API are independent of the actual messaging protocol used between sender and recipient.

The eMessage Connector is defined between an LN messaging-enabled application and one or more service providers. On the client side, communication is one-directional, from LN to external service providers. On the server side, communication can occur in both directions.

The overall process is depicted in this diagram:



Definitions

These definitions are used:

4GL process
The 4GL process can be any LN 4GL process that must communicate with another process not located in the same LN Virtual Machine (BVM). This external process can be another LN process running on a separate BVM or third-party messaging-based software. This 4GL process can send messages to an external process and it can process incoming messages.

eMessage Connector DLL

The eMessage Connector DLL is a 4GL DLL that any LN process can use to create, send, and receive message objects. The DLL contains functions used to create, examine, and transport message objects. The DLL accesses the eMessage Connector repository to determine the available message types and the connectors to handle those message types. When an LN process wants to send a message, the eMessage Connector DLL examines the repository to determine and starts the appropriate connector required to handle the message.

eMessage Connector

The eMessage Connector acts as the intermediary between the 4GL process and the service provider. When the connector is started, it determines from the eMessage Connector repository how to connect to the external provider. Then a connection is established. From now on the connector will handle the transport of the message objects between the 4GL process and the external provider.

The connector also performs message logging. All messages passing through the connector are logged to the eMessage Connector repository log table and the individual message parts are stored in the file system (if logging is enabled). Storing message objects as files ensures traceability and is also important for LN 4GL applications. The LN 4GL does not allow the allocation of large memory areas for processing BLOBs (Binary Large Objects) and other large objects. The message log therefore allows the LN applications to digest these large files in smaller chunks, by reading them from the file system in small pieces.

eMessage Connector repository

The repository stores several types of information about the eMessage Connector environment:

- eMessage Connector services: supports email.
- Message types supported per service: SMTP
- eMessage Connector Service providers: supports SMTP
- Message logs

Message handling

Outbound message handling

An LN process completes several steps to send a message to a service provider:

- The LN process uses the eMessage Connector DLL functions to create the message.

- The LN process instructs the DLL to use a particular service to send the message. A service is associated with a connector, which provides the actual service.
- The DLL accesses the eMessage Connector repository to determine which connector to use to process the message object.
- The eMessage Connector starts this connector and passes the message object to it.
- The connector accesses the repository to determine the appropriate parameters to be used to connect to the provider. Connection is established to the service provider and the message is sent.
- The eMessage Connector performs the appropriate message logging.

eMessage Connector message structure

eMessage Connector messages are represented internally as LN Virtual Machine message objects. On disk, the messages are represented as a set of one or more files. The first file, known as the header, is an XML document. If other files exist, they are sent as attachments to the XML header and specified as such in the header. The format of these attachments is application-specific and depends on the interface agreement between the applications that exchange these files. The transmission method of eMessage Connector files between the connector and the provider is provider-specific.

Chapter 2: Repository

How LN uses the repository

The repository holds information about the services and service providers that the eMessage Connector can use. You can use the repository sessions to enter and update services and services providers.

When LN receives a message, it first determines the type of message service required. LN uses the **Address Types by Service (ttcmf0140m000)** session for this.

After LN knows what type of message service to use, it refers to the Services (ttcmf0130m000) session to determine what type of service provider to use. LN then uses the **Service Providers (ttcmf0110m000)** session to link the message to the correct 4GL connector.

Before you can use eMessage Connector, you must specify this information:

- LN output devices, to support eMessage Connector in job mode
- Message services
- Service providers

LN output devices

SMTP output device

Use the **Device Data (ttaad3500m000)** session to create an SMTP device.

To create an SMTP device:

- Add a device and specify a description of this device.
- In the **Device Type** field, select **`Rewrite File`**.
- Specify the system locale in the **Locale** field.
- In the **File** section, specify **`ttcmfprint`** in the **4GL Program** field.
- Specify **`smtp`** in the **Argument** field.
- Specify the default file location for temporary files in the **Path** field.
- Specify the default value, **`66`**, in the **Page Length** field.

The repository sessions

Service Providers (ttcmf0110m000)

Use the **Service Providers (ttcmf0110m000)** session to link each connector to a particular provider. The value that you specify in the **4GL Connector** field must exist in the **Sessions (ttadv2500m000)** session.

The session contains this command:

Provider Parameters (Ctrl+Shift+P)
This command starts the **Provider Parameters (ttcmf0120m000)** session to maintain the parameters for the provider.

To specify a service provider:

1. In the **Provider** and **Description** fields, specify the name and description of the new service provider.
2. In the **4GL Connector** field, specify the session name of the LN 4GL connector process associated with the provider.

   You can zoom to the **Sessions (ttadv2506s000)** session.

Currently eMessage Connector only supports SMTP as service provider.

Provider Parameters (ttcmf0120m000)

You can start the **Provider Parameters (ttcmf0120m000)** session from the **Service Providers (ttcmf0110m000)** session. Use the session to specify the startup parameter values for a particular connector.

The eMessage Connector automatically fills the provider parameters using the 4GL Connector when the **Provider Parameters (ttcmf0120m000)** session is started. You cannot add parameters, but you can change existing parameter values in the **Parameter Value** field.

In case of connection issues, check the files in these folders:

- `$BSE/log/logfiles`
- `$BSE/tmp/bshell-trace`

Field	Description
No.	The order of the parameters. This field is automatically filled.
Parameter Description	A description of the parameter. This field is automatically filled.
Parameter Value	The value associated with the particular parameter.

SMTP provider parameters

The SMTP provider is linked to one of these connectors:

- ttcmfsmtp
- ttcmfcurlsmtp

  ttcmfcurlsmtp is similar to ttcmfsmtp, but it uses the underlying cURL. cURL is part of the porting set.

ttcmfsmtp

If the ttcmfsmtp connector is used, you must specify these parameters to configure the SMTP provider:

- The **SMTP mail host**.
- The **SMTP port**.
- The **Connection timeout** in seconds.
- The **<Mail From> domain**.
- **SMTPs** (secure SMTP): select **`Yes`** or **`No`**. If you select **`Yes`**, the **Mail account** and **Mail Password** parameters are mandatory to specify. Leave the **<Mail From> domain** parameter blank.
- The **Mail account**.
- The**Mail password** of the mail account. On the appropriate menu, select **Add Password**.

ttcmfcurlsmtp

If the ttcmfcurlsmtp connector is used, you must specify these parameters to configure the SMTP provider:

- The **SMTP mail host**.
- The **SMTP port**.
- The **Connection timeout** in seconds.
- **SSL (CURL)**: Select **`Yes`** or **`No`**. If you select **`Yes`**, the **Mail account** and **Mail Password** parameters are mandatory to specify.
- The **Mail account**.
- The**Mail password** of the mail account. On the appropriate menu, select **Add Password**.

Services (ttcmf0130m000)

Use the **Services (ttcmf0130m000)** session to specify the available services and the providers that support them. Only one provider per service per environment is allowed.

The message storage path directory and the inbox and outbox subdirectories are created after you save each entry.

To specify a service and link it to a provider:

1 Specify the name and description of the service.

2 Specify the name of a provider from the **Service Providers (ttcmf0110m000)** session. Click the right arrow to display a list of valid providers.
3 Select **Enabled** to enable the service. A service must be enabled before it can be used. Clear **Enabled** to disable the service. For example, when a service provider is not available.
4 Specify the path where messages from this service are stored. Inbox and outbox subdirectories are created under this directory. Select **Logging** to enable logging.
5 Choose the default attachment file type for the service. When LN sends a report using eMessage Connector, the report is converted to the format specified here.

Field	Description
Service Name	Specify the name of the service offered by a provider.
Description	Specify a description of the service.
Provider	Specify a provider from the **Service Providers (ttcmf0110m000)** session. You can click the right arrow to display a list of valid providers.
Enabled	Select to enable the service. A service must be enabled before it can be used.
Logging	Select if you want messages sent by this service to be logged to the message log, which is displayed in the **Message Log (ttcmf1500m000)** session.
Message Storage Path	Specify the path where messages from this service are stored. Inbox and outbox subdirectories are created under this directory.
File Type	Select the preferred attachment type for the service. When LN uses eMessage Connector to send a report, the report is converted to this format. Select one of these options from the drop-down list: • Plain Text ASCII • Postscript • RTF • HTML ASCII • HTML Unicode • Plain Text Unicode • PDF

Address Types by Service (ttcmf0140m000)

Use the **Address Types by Service (ttcmf0140m000)** session to set up the link between an address type and its type of service.

To link an address type to a service:

1 Specify an address type and description.
2 Specify the name of the service. Each service can have more than one address type. Therefore, a service can be displayed on this list more than once.

For SMTP two address type configurations are supported.

Field	Description
Address Type	Specify a valid eMessage Connector address type, such as smtp. There are no restrictions on the content of this field, so client sites can customize the address types.
Description	Specify a description of the address type.
UI Required	Select to allow additional user interaction before the message transfer is complete.
Service Name	Specify a service from the **Services (ttcmf0130m000)** session. Click the right arrow to view a list of valid services. Each service can have more than one address type.
Paper Type	Select the paper type. You can zoom to the **Paper Types (ttaad3100m000)** session.

Chapter 3: Message log

Tracking messages

The eMessage Connector logs all incoming and outgoing messages as they are processed. You can use the message log sessions to view these messages. You can view the messages grouped by LN message ID, by service name, or by application name.

The message log sessions

Message Log (ttcmf1500m000)

The **Message Log (ttcmf1500m000)** session contains a log of all the processed messages, and their status. The LN message ID corresponds to the subdirectory name in which the message parts are stored, so that you can retrieve the actual message. External messages are assigned an LN message ID in addition to retaining their external ID.

You can sort messages according to the time stamp, LN ID, or external ID.

The Specific menu contains this option:

Delete Messages (Ctrl+Shift+D)
Starts the **Delete Message Log (ttcmf1200m000)** session.

To view messages in the message log:

1. Double-click a message to see the message details. The **Message Log Details (ttcmf1500s000)** session contains tabs for **Identification** and **Delivery** tags. They are displayed on two form tabs.
2. Click **Recipients** to see a list of recipients.
3. Click **Attachments** to see a list of attachments. Select **Specific > Edit/View Attachments** to see the details of one attachment.
4. You can use the **Print Messages** option on the **File** menu to print a range of messages in their full serialized XML form.

Field	Description
Infor LN Message ID	Shows the unique message ID of the message in LN. The Infor LN message ID corresponds to the directory name where the serialized message components are stored. The message ID is a 40 character string max. The first 10 characters is the UTC date and time (number of seconds since Jan 1, 1970 converted to a string) followed by a ".". The next five characters are the bshell pid (mod 99999) followed by a ".". The next three characters are the bshell process ID (mod 999) followed by a ".". The next three characters are a counter in case of multiple transactions per second (max 999) followed by a ".". The last characters are the first few characters of the machine's host name.
Application Name	Shows the LN session name of the application that sent or handled the message.
Service Name	Shows the service name from the **Services (ttcmf0130m000)** session. The application that is displayed in the **Application Name** field controls which eMessage Connector services it handles.
Time Stamp	Shows the date and time (UTC) that the message was originally obtained by eMessage Connector.
External/Original ID	Shows the unique message ID of the message in the external system. This is only filled if the message originated in the external system.
Status	Shows the current status of the message, from the perspective of the connector. The possible values are: • Message Received • Message Sent • Message Error • Reply Received • Reply Sent • Reply Error
Status Time	Shows the time of the status being reported.
Status Message	Shows further information about the status, for example, an error message or the message ID of a reply received.

Delete Message Log (ttcmf1200m000)

You can delete messages by specifying ranges on the LN ID, external ID, service name, application name and time stamp. When you delete a message, the database entry and all associated disk files are removed.

Message Log by Service Name (ttcmf1501m000)

You can use the **Message Log by Service Name (ttcmf1501m000)** session to view the messages grouped by service name. This session has the same functionality as the **Message Log (ttcmf1500m000)** session.

For information about the fields, see Message Log (ttcmf1500m000) on page 14.

The **Message Log by Service Name (ttcmf1501m000)** session has one additional field. **Message Direction** shows whether the message was inbound (to the bshell) or outbound (from the bshell). Inbound messages are stored in the inbox subdirectory of the message storage directory. Outbound messages are stored in the outbox subdirectory of the message storage directory.

Message Log by Application Name (ttcmf1502m000)

You can use the **Message Log by Application Name (ttcmf1502m000)** session to view the messages grouped by service name. This session has the same functionality as the **Message Log (ttcmf1500m000)** session.

For information about the fields see Message Log (ttcmf1500m000) on page 14.

The **Message Log by Application Name (ttcmf1502m000)** session has one additional field. **Message Direction** shows whether the message was inbound (to the bshell) or outbound (from the bshell). Inbound messages are stored in the inbox subdirectory of the message storage directory. Outbound messages are stored in the outbox subdirectory of the message storage directory.

Chapter 4: Addressing

Storing address information

The address book makes it easier to retrieve addressing information about users. Currently, user information is maintained in several areas of the LN application. The eMessage Connector Address Book stores addressing information in a central location. This allows applications to both display a list of addresses and to determine the address for a particular recipient.

The data access layer (DAL) object for the address book contains functions to add, modify, and delete recipients' information. LN applications also use the DAL functions to update the address book whenever relevant information changes.

The DAL functions perform property checks on fields where possible, for example, SMTP addresses (check for @ sign), default address (if specified, an address of that type must be present). An application program can update the address book by calling `dal.set.property()` for the fields to update, and then using `dal.new()`, `dal.destroy()` or `dal.update()` as appropriate. Ensure that your application sets the DAL property `Change Allowed` to `No` if you do not want to change particular entries in the address book.

You can also build distribution lists from the entries in the address book. The distribution lists makes it easier to set up multirecipient messaging.

The Addressing sessions

Address Book (ttcmf2100m000)

The **Address Book (ttcmf2100m000)** session allows you to add and update entries in the address book. Use the main session to add an entry and use the details session to specify information or update the entry.

You cannot delete an entry if it is present in other tables. If the entry can be deleted, the DAL removes it automatically from all distribution lists. When you double-click an entry in the Distribution List category, the **Distribution Lists (ttcmf2110m000)** session is displayed, displaying the members of the selected list.

The **Specific** menu contains this option:

Distribution Lists (Ctrl+Shift+D)

Starts the **Distribution Lists (ttcmf2110m000)** session.

To enter a new address book entry:

1. Select **File > New**

   The **Address Book Details (ttcmf2100s000)** session is displayed.
2. Specify the category of this entry. This is a free form text field with no constraints. You can develop your own categories or use one of the supplied categories such as Employee, Business Partner, or LN User. Click **Zoom** to view the available categories for category LN User.
3. Specify a new key, or tab through this field to accept the key LN generates. This key corresponds to the user listed in the User Data table for the LN User category. For other categories other keys can be used as long as the combination category/key is unique.
4. Specify the address type of the email address. Only SMTP is supported.
5. Specify the email address of the recipient.
6. Specify a default address type for the user.

Field	Description
Key	You can also enter a new key. Note that the combination category/key must be unique.
Display Name	Specify or let LN enter the full name of the recipient to which the entry applies, for example, John Smith.
Default	Specify the default address type for the user. LN uses the default when an application does not specify how a message must be sent to the user. If you use a default message type, you must also supply an address for that type. You can also leave this field empty, meaning this user has no default message type.
Category	Specify the category of the entry. This is a free form text field with no constraints. You can develop your own categories. The initial categories are Employee, Business Partner, LN User, and Distribution List. The category Distribution List is reserved and you cannot add records to this category. You can zoom to the **Categories (ttcmf2501m000)** session.
E-mail Type	Specify the address type of the email address. Only SMTP is supported.
E-mail Address	Specify the email address of the recipient. A validity check is run on the email address based on the email type.

These fields are displayed in the session, but not yet supported:

- Telephone Number
- Fax Number
- Telex Address
- SITA Address
- SMS Address

These internal fields are available by way of an application:

- The **Change Allowed?** field indicates whether you can modify the current record. When an application adds an entry to this table, the application sets this field to **No** to prevent users from modifying the entry. This keeps the address book synchronized with the application table. When a user makes a manual entry, this field is automatically set to **Yes**. Use domain `ttyeno` for this field.
- **Cmba** is a combination field with child fields category and key. This allows you to build a reference link to this table from the distribution list.

Distribution Lists (ttcmf2110m000)

The **Distribution Lists (ttcmf2110m000)** session enables you to create and maintain distribution lists. You can also create nested distribution lists.

The **Specific** menu contains this option:

Address Book (Ctrl+Shift+A)
Starts the Address Book (ttcmf2100m000) session.

To create a distribution list:

1. Click **New Group** and specify a name for this list.
2. In the **Display Name** field, select a recipient from the **Address Book (ttcmf2100m000)** session. Click the right arrow to see a list of valid recipients.
3. Select an address type from the drop-down list. You can choose more than one address type for each recipient.
4. Continue adding recipients until the distribution list is complete.
5. To create a nested distribution list, create a new distribution list. In the **Display Name** field, specify the name of the distribution list that you want to include. Continue adding recipients until the distribution list is complete.

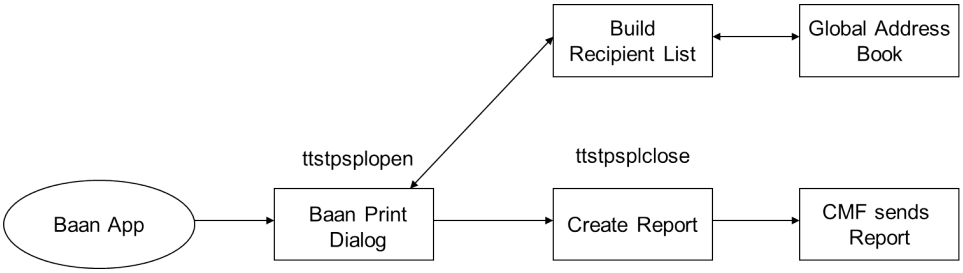
Field	Description
List Name	Specify the name of the distribution list.
Display Name	Specify the name of a recipient from the **Address Book (ttcmf2100m000)** session. Click the right arrow to see a list of valid recipients.
Address Type	From the drop-down list, select the address type of the recipient to be used.
Category	Shows the category as specified in the **Address Book (ttcmf2100m000)** session.
Key	Shows the unique ID of an entry within a particular category. The value of this field is application-dependent.

Cmba is an internal field that you can use to construct a restricted reference to the Address Book table. When an entry is deleted from the address book, it is also deleted from all distribution lists.

Chapter 5: To print reports

Architecture

This diagram shows the architecture of CMF Printing:



The preceding diagram shows the various components in the CMF Printing architecture:

Baan App
This is the LN application (along with the LN Standard program) that selects the data for the LN report.

Baan Print dialog box
This dialog box is extended with a CMF-specific tab, which allows you to compose the message header required by CMF. Among other things, you can construct a recipient list (containing TO, CC, BCC and OBO recipients) and the message subject. You can also enter free form text to be sent along with the LN report by clicking the message button.

Build Recipient List (ttcmf2120m000) (messages)
This session is used to build a recipient list for the message. You can fill in the to, cc, bcc, obo and notify recipients here.

Global Address Book
This dialog is used by the CMF Message Dialog for composition of the recipient lists.

Create Report Object
This is the standard report object that produces the LN reports for a particular LN session.

Baan Report
This is the file that is the output of the LN report object.

Baan Report Object Close
This process performs the various cleanup activities required when handling reports. It is a modified version of the standard report close object. It composes a CMF message object based on the output

of the Baan report object and the print dialog then activates the necessary CMF service and passes the object to it.

The CMF Print dialog box

The information that you specify in this session is used to build an eMessage Connector message object without a recipient list.

The ID of the message object is stored in the `cmf.MessageObject bic_global` variable. The recipients in the **Recipient List** session are converted to an address list object, and the ID is stored in the `cmf.AddressList bic_global` variable. Both of these variables are exported to the calling LN program when the Print dialog box ends are then available to the LN application.

If you leave the recipient list blank, the service listed in the **Address Types by Service (ttcmfm000)** session are used. The first service found with **UI Required** (in table ttcmf040) enabled with the correct address type is used to display the message (with the report as attachment) to the user. The user can add some recipients and send the message.

A separate Message Dialog program, similar to the **Message** tab, allows other sessions besides the Print dialog box use the Send Message tab. Any eMessage Connector application can use this Message Dialog program to interactively construct the message header.

The Message dialog program creates Task and Appointment objects using information entered by the user on the appropriate tabs and the task and appointment functions described in the *Programmer's Manual*.

Application programmers can access the functionality of the Message dialog program by use of the `cmf.MessageDialog ()` function call. This function is fully described in the *Programmer's Manual*. Programmers must note that the address list and message objects used by the Message Dialog program are not destroyed upon return from this function. It is the programmer's responsibility to handle the destruction of these objects (the IDs of these objects are stored in the `cmf.AddressList` and `cmf.MessageObject) (bic global)` variables.

Both the Baan Print Dialog and the Message Dialog sessions must check the **Address Types by Service** table at start up. Only the address types listed in this table are allowed in the **Recipient List (ttcmf2120m000)** (messages) session.

LN Report Object Close

This program completes the sending of the message. It retrieves the message object and address list created by the Message Dialog by importing these from the calling LN program. It then calls the `cmf.sendtoPerson()` command to deliver the message. One of the arguments to `cmf.send.toPerson()` is the name of the Baan Report file.

This `cmf.sendtoPerson` file converts the Baan Report into the default format, specified in the **Services (ttcmf0130m000)** session, for each service to which it sends the message. After `cmf.sendtoPerson` is finished, `cmf.stopAllServices()` is called to close all of the open CMF services.

If the message delivery failed for any of the recipients, then this session shows an error message containing the list of recipients for which delivery was unsuccessful. If message delivery worked for all recipients, a success message is displayed.

The address list object can be empty when this session is invoked. This implies that you want to use a UI-based CMF service to perform the actual addressing.

The Print dialog box

You can print reports through eMessage Connector from any print session in Infor LN. The **Send Message** tab on the **Print** dialog box contains fields to identify the message. Specify delivery options for the message, and specify information about the recipients.

Field	Description
Message Identification	
Subject	Specify the subject of the message.
Categories	This field contains a comma-delimited list of the categories to which the message belongs.
Delivery Options	
Priority	Specify the priority of the message. The possible values are Low, Normal, or High. The default is Normal.
Sensitivity	Specify the sensitivity of the message. The possible values are Normal, Personal, Private, Confidential, and Secret. The default is Normal.
Deliver After	Specify after what time the message must be delivered. If you leave this field empty, the message is delivered immediately.
Expire After	Specify what time the message must expire. If you leave this field empty, the message never expires.
Notification	Specify the notification options for the message. The possible values are Always, Delivery, Non Delivery, Never. The default is Non Delivery. Notifications are never sent by the client-side service.
Show message before sending	Select to preview the message before it is sent. This can be used for a client-side service.
Fax Options	

Field	Description
Template Code	This field is not yet supported by LN.
Charge Code	This field is not yet supported by LN.

The message ID and message create time are automatically handled by eMessage Connector.

Sending reports through SMTP to one or more recipients

You can send processed reports directly to one or more recipients:

- Click the **Send Message** tab in the **Select Device (ttstpsplopen)** session.
- Specify the mail subject in the **Subject** field.
- Clear the **Show message before sending (if possible)** option.
- Specify the recipients information in the **Recipients** sub session.
- Click **Message**. Specify the message in the text editor session.

Sending reports through SMTP to a one recipient

You can send processed Baan reports from the Baan Server through SMTP to one recipient. This is configured in the Select Device dialog on the **File** tab. This method requires a preconfigured SMTP device in LN.

- Specify an SMTP-configured device in the **Device** field.
- Specify the recipient's mail address in the **Output file** field.

The recipient receives an email with an attachment. The **From** field contains the description and code of the processed report. The format of the attachment is defined in the **Addresstypes by Service (ttcmf0140m000)** session.

Sending report through SMTP in job mode

You can process a report through a job. Follow the standard procedure to create a job and specify this information in the **Add Session to Job (ttaad5102s000)** session:

- In the **Device** field, specify the device that is configured for SMTP.
- In the **Output file** field, specify the recipient's email address.

Recipient List (ttcmf2120m000) (messages)

With the **Recipient List (ttcmf2120m000)** (messages) session you can build a recipient list for the message to be sent.

Use this session to build a recipient list by zooming to an address book or by entering the information manually. If you change the address type for a recipient in the list, the address book is accessed and the address corresponding to the new type is placed in the **Address** field. If no address is found for the new address type, the address field remains clear.

The variable `cmf.AddressList` is declared as a long in the `bic_global` LN include. On startup this session will import the value of this variable. The idea is that if the calling session already has a valid address list, then the display session will add any further addresses (also the addresses in the imported address list are displayed) you select to the already existing address list object.

If no current valid address list object is available, the display session will create a new one with the addresses you select and return in `cmf.AddressList`.

The fields of the **Recipient List (ttcmf2120m000)** (messages) session are listed in this table:

Field	Description
Role	This is the role of the recipient. The possible values allowed here are TO | CC | BCC | OBO. The From role is not allowed. The From value is determined from the LN User information for the current user.
Name	The display name of the recipient is entered here. You can optionally zoom to the address book to get a recipient from there.
Type	This field contains the address type of the recipient. You can only enter address types here that are present in the Address Types by Service CMF table. In other words, only the address types that are supported and implemented at the customer site can be used here.
Address	This field stores the address of the recipient. If you zoomed to the address book, this is automatically filled; otherwise you can manually specify the information here.

Recipient List (ttcmf2121m000) (tasks)

With the **Recipient List (ttcmf2121m000)** (tasks) session you can build a recipient list for a task to be sent. It is essentially the same session (with the same functionality) as the session described in Recipient List (ttcmf2120m000) (messages) on page 23. Only the **Role** field is replaced by a **Status** field and there is no **Address Type** field. This is not required because tasks and appointments can only be sent to the email address type.

The fields of the **Recipient List (ttcmf2121m000)** (tasks) session are listed in this table:

Field	Description
Status	This is the status of the recipient. The possible values are `To` and `Update`. Recipients with status `Update` are placed on the update list of this task.
Name	The display name of the recipient. You can optionally zoom to an address book to get a recipient from there.
Address	The email address of the recipient. If you select a distribution list as a recipient, the Address field is clear.

Attendee List (ttcmf2122m000)

The **Attendee List (ttcmf2122m000)** session allows you to build a list of attendees for an appointment to be sent. The functionality of this session is similar to the sessions described earlier.

The session contains these fields:

- **Attendance**

  This is the attendance status of the recipient. The possible values allowed here are **Required Attendee**, **Optional Attendee**, **Resource**.
- **Name**

  The display name of the recipient is entered here. You can optionally zoom to an address book to get a recipient from there.
- **Address**

  The email address of the recipient.

If you select a distribution list as a recipient, the **Address** field is not available.