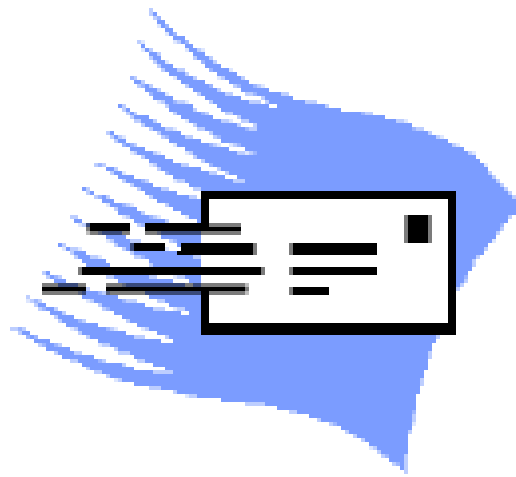


Communiqué



User Guide

*Version V1R1M0
November 2003*

Table of Contents

Introduction	4
Why Choose Communiqué?	4
Prerequisites	5
Important Note about POP settings!	5
Address Lists	6
Installation	7
Method 1: Installing Communiqué using FTP	7
Method 2: Installing Communiqué using the IFS	9
Maintenance	12
Method 1: Installing a PTF using FTP	12
Method 2: Installing a PTF using the IFS	14
Using Communiqué	16
The Communiqué Menu	16
Sending a Communiqué Email Message	17
TO	17
FROM.....	19
SUBJECT	21
MSG	21
ATTACH.....	23
PRIORITY	26
CFMDEL.....	26
SNDMLTMSG	26
Working with Address Lists	27
Creating an address list	29
ADRL.....	29
TEXT	29
Changing an address list	30
ADRL.....	30
TEXT	30
Copying an address list	31
FROMADRL	31
TOADRL.....	31

TEXT	31
CPYADRLE	31
Deleting an address list	32
ADRL	32
Converting a Distribution List to an Address List.....	33
DSTL	33
ADRL	33
TEXT	33
HOSTNAME	33
DOMAIN	33
Working with Address List Entries	35
ADRL	35
Adding an address list entry	37
ADRL	37
EMAIL	37
NAME	37
TYPE	37
Changing an address list entry	38
ADRL	38
EMAIL	38
NAME	38
TYPE	38
Copying address list entries	39
FROMADRL	39
TOADRL	39
Removing an address list entry	40
ADRL	40
EMAIL	40
Using the Communiqué API CM_SNDAPIR	41
Required Parameter Group	42
The Communiqué Database	46
Message Log CM_MSGLOG.....	46
Message Recipients CM_MSGRCP	46
Message Attachments CM_MSGATC	46

Introduction

Communiqué provides iSeries users and applications with easy to use but fully functioned facilities for sending email messages in MIME format.

With **Communiqué** you can:

- send email to Internet email users as well as email users on the same iSeries or another iSeries
- send email messages in simple text or HTML or alternative text/HTML formats
- attach or embed text, HTML, image, video, audio or application binary files (for example, PDF files and Excel spreadsheets)
- use a simple command interface or an API
- create and manage named email address lists
- send to multiple email addresses or to named email address lists
- convert SNADS distribution lists to email address lists

Communiqué also dovetails with @riadne's other products, **CoolSpools** and **Slipstream**. Installing **Communiqué** will enable the email capabilities inherent in those products (requires **CoolSpools** Version 4.1 and **Slipstream** Version 3.1. Once **Communiqué** is installed, using the **CoolSpools** email-related parameters will allow you to convert a spooled file to a stream file in PDF or a variety of other formats and send that stream file as an email attachment. Similarly, if **Communiqué** is installed, the **Slipstream** email-related parameters will enable you to convert a database file to an Excel spreadsheet or other stream file format and email that stream file as an attachment.

Why Choose Communiqué?

There are many PC packages which enable you to send email. There are also packages that let you to send email from your iSeries using a GUI. So why choose **Communiqué**?

Communiqué provides a means of sending email from an iSeries command. That may not be as pretty as a GUI, but it's a darn sight easier to integrate with your existing applications.

Just think about it. How is a GUI going to help you convert your billing application to let you send invoices out to customers by email? With **CoolSpools** and **Communiqué** it can take little more than adding a few command calls to your CL program to convert your invoice spooled files to PDF format and email them.

To take another example, what if you want to email your internal users their sales report in Excel format each morning rather than printing it and distributing it on paper? A GUI is not much help there. With **Slipstreams** and **Communiqué**, you can run your query and save the output as an Excel file rather than a spooled file and then email the file as an attachment, all with a couple of simple program changes to call the appropriate commands.

One final idea for you: do you have to support iSeries batch jobs that run overnight or at weekends? Why not put a call to **Communiqué** in your program error handling routines? That way, when an error occurs, you can receive an email at home or in the office. Neat.

Prerequisites

Communiqué provides means of creating email messages in MIME (Multipurpose Internet Mail Extensions) format on the AS/400 or iSeries. For these email messages to be routed and delivered successfully, you must have previously configured and activated the appropriate email environment on your system. Furthermore, if the email is to be delivered to a user on another system, or to an Internet user, then the necessary TCP/IP communications connections must already have been put in place.

A number of IBM documents in PDF format are available for downloading from the @riadne website to assist you with configuring email on your system at www.ariadnesoftware.co.uk/email.htm.

For further information on how to set up your iSeries for email, consult the following IBM links:

<http://publib.boulder.ibm.com/series/v5r1/ic2924/info/rzair/rzairgetstart.htm>

http://www-912.ibm.com/s_dir/SLKBase.nsf/3ff1161e61e6c7468625661300765a3f/34879f5e413aa5cd8625697b0053f36a?OpenDocument

OS/400 V4R5M0 or higher.

5 Mb of iSeries disk space.

No PC is required.

For **CoolSpools** integration: **CoolSpools** Version 4.1

For **Slipstream** integration: **Slipstream** Version 3.1

Important Note about POP settings!

Prior to OS/400 V5R3M0, the default for the POP MSGSPLIT attribute was set to a value which meant that large email attachments were split across multiple messages. This often resulted in those email attachments becoming corrupted by the AS/400 email servers after they had been created by Communiqué.

We recommend strongly that you set your MSGSPLIT level to *NOMAX to prevent this happening, e.g.:

CHGPOPA MSGSPLIT(*NOMAX)

If you are experiencing problems with large attachments getting corrupted, this is almost certainly the solution.

Address Lists

Communiq  implements the concept of a named **Email Address List**. Address lists can be extremely useful when you need to send messages to groups of people on a regular basis. For example, you might create an address list for each department within your organization (e.g. SALES, MARKETING, IT etc.). Then, when you need to send a message to everyone in the Sales Dept., you just specify the **SALES** address list as a recipient for the message.

Communiq  provides a set of tools for creating and managing address lists and their contents.

Installation

If you downloaded **Communiqué** from the Internet, or received the product via e-mail, you will have received a zip file called **CMNQUEV1R1.ZIP**. This zip file contains everything you need to get started with **Communiqué**. This section describes two possible procedures for installing **Communiqué** on your AS/400. We are hoping to simplify and automate this area in the near future. However, for the moment, please select one of the two methods described below and follow the steps carefully and in the order indicated. Steps to be carried out on the PC are shown in **green**, steps to be carried out on the AS/400 in **blue**.

Method 1: Installing Communiqué using FTP

This method uses FTP (File Transfer Protocol) to transfer the Communiqué product files from your PC to your AS/400. It is assumed that you have stored the CMNQUEV1R1.ZIP file which you downloaded from the Internet or received via e-mail on your PC and that you have enabled an FTP link between your PC and your AS/400.

Step 1

On your PC, create an install directory for Communiqué, either within Windows Explorer, or from the DOS prompt, e.g.:

```
md c:\CMNQUEV1R1
```

Save or copy the zip file that you downloaded or received into this directory.

Step 2

On your PC, unzip the contents of the zip file into the directory you just created using pkunzip, WinZip, Norton UnZip etc.

Step 3

On your AS/400, decide which library you wish to use for the install procedure and create a save file in that library, e.g.

```
crtssvf lib_name/CMNQUEV1R1
```

where **lib_name** is the name of the library you chose to use.

Step 4

On your PC, start an FTP session to your AS/400. You may use FTP packages such as CuteFTP or simply go to the DOS prompt and type:

```
ftp AS400_name
```

where AS400_name is the name of your AS/400.

You will be prompted to enter your user id and password. Please ensure that you sign on with a user id that has adequate authority. We recommend that you sign on as the Security Officer QSECOFR.

Issue the following FTP commands within the FTP session:

```
binary
```

This command instructs FTP not to carry out any conversion on the data to be sent to the AS/400.

```
cd lib_name
```

where lib_name is the name of the library in which you created the save file in Step 3. This command makes lib_name your current library on the AS/400.

lcd dir_name

where dir_name is the name of the directory on your PC in which you unzipped the install files earlier. This command makes dir_name your current directory on the PC.

put CMNQUEV1R1.sav CMNQUEV1R1

This command transmits the contents of the file CMNQUEV1R1.sav, which was earlier unzipped from CMNQUEV1R1.zip, and stores them in the save file called CMNQUEV1R1 which you created earlier on your AS/400.

Step 5

On your AS/400, ensure that the previous step worked by displaying the contents of save file CMNQUEV1R1:

dspsavf lib_name/CMNQUEV1R1

If you see the message (**No records in save file**) then something has gone wrong during the preceding steps. Try following them again. If you still cannot get this procedure to work, contact us at support@ariadnesoftware.co.uk.

Step 6

On your AS/400, issue the following command to restore the Communiqué licensed program to your system:

```
rstlicpgm      licpgm(1CMNQUE)  
               dev(*savf)  
               option(*base)  
               rstobj(*all)  
               lng(*primary)  
               output(*none)  
               rls(*first)  
               replacerls(*only)  
               savf(lib_name/CMNQUEV1R1)
```

where lib_name is the name of the library in which you created the save file in Step 3.

Step 7

If you have downloaded Communiqué from the Internet, you can use the product without a license key for up to 30 days. At the end of that period Communiqué will notify you that it can no longer be used without a license key.

To obtain a license key contact ariadne at sales@ariadnesoftware.co.uk.

If you have been issued with a license key, proceed as follows.

On your AS/400, issue the following command to add a license key to enable you to use the Communiqué licensed program:

```
addlickey      lickeyinp(*prompt)  
               prdid(1CMNQUE)
```

**licterm(V3R1)
feature(5001)
serial(*local)
prcgrp(Ppp)
lickey(XXXXXX YYYYYY ZZZZZZ)
usglmt(*nomax)
expdate(exp_date)
vnddta(*none)**

where: **XXXXXX YYYYYY ZZZZZZ** is the license key you received by e-mail from
US
exp_date is the expiry date mentioned in the e-mail
and: **Ppp** is the processor group mentioned in the e-mail

Communiqué is now ready for you to use!

If you have any trouble at any point in this procedure, please contact us by e-mail at
[**support@ariadnesoftware.co.uk**](mailto:support@ariadnesoftware.co.uk).

Method 2: Installing Communiqué using the IFS

This method uses a Client Access network drive assigned to the IFS of your AS/400 to transfer the Communiqué product files from your PC to your AS/400. It is assumed that you have stored the CMNQUEV1R1.zip file (which you downloaded from the Internet or received via e-mail) on your PC.

Step 1

On your PC, create an install directory for Communiqué, either within Windows Explorer, or from the DOS prompt, e.g.:

```
md c:\PC_dir
```

where PC_dir is the name of the directory you wish to use for this installation.

Save or move the zip file that you downloaded or received into this directory.

Step 2

On your PC, unzip the contents of the zip file into the directory you just created using pkunzip, WinZip, Norton UnZip etc.

Step 3

On your PC, ensure that you have a network drive assigned to a directory in the root file system of your AS/400's IFS. This can be done in Windows Explorer by selecting Tools, Map Network Drive.

Create a directory in the IFS to receive the Communiqué files. This can be done in Windows Explorer by selecting File, New, Folder, or at the DOS prompt as follows:

```
md I:\IFS_dir
```

where **I:** is the network drive assigned to your AS/400 and IFS_dir is the name of the directory you wish to use for this installation.

Step 4

Copy the contents of your PC's install directory created in Step 1 above to the IFS directory created in Step 3 above using Windows Explorer, or from the DOS prompt as follows:

```
copy c:\PC_dir\*. * I:\IFS_dir
```

where **PC_dir** is the name of the directory on your PC, **I:** is the network drive assigned to your AS/400 and **IFS_dir** is the name of the directory you wish to use for this installation.

Step 5

On your AS/400, decide which library you wish to use for the install procedure and create a save file in that library, e.g.

```
crtsave lib_name/CMNQUEV1R1
```

where **lib_name** is the name of the library you chose to use.

Step 6

On your AS/400, copy the contents of the CMNQUEV1R1.sav file from the IFS directory to the save file just created, by issuing the following command:

```
copyfrmstmf      fromstmf('/IFS_dir/CMNQUEV1R1.sav')  
                  tombr('/qsys.lib/lib_name.lib/CMNQUEV1R1.file')  
                  mbropt(*replace)  
                  cvtdta(*none)  
                  endlinfmt(*fixed)  
                  tabexpn(*no)
```

where **IFS_dir** is the name of the IFS directory into which you copied the install files in Step 4 above, and **lib_name** is the name of the library in which you created the save file in Step 5 above.

Step 7

On your AS/400, ensure that the previous step worked by displaying the contents of save file CMNQUEV1R1:

```
dspsavf lib_name/CMNQUEV1R1
```

where **lib_name** is the name of the library in which you created the save file in Step 5.

If you see the message **(No records in save file)** then something has gone wrong during the preceding steps. Try following them again. If you still cannot get this procedure to work, contact us at support@ariadnesoftware.co.uk.

Step 8

On your AS/400, issue the following command to restore the Communiqué licensed program to your system:

```
rstlicpgm      licpgm(1CMNQUE)  
                dev(*save)  
                option(*base)  
                rstobj(*all)  
                lng(*primary)  
                output(*none)  
                rls(*first)  
                replacerls(*only)
```

savf(lib_name/CMNQUEV1R1)

where lib_name is the name of the library in which you created the save file in Step 5.

Step 9

If you have downloaded Communiqué from the Internet, you can use the product without a license key for up to 30 days. At the end of that period Communiqué will notify you that it can no longer be used without a license key.

To obtain a license key contact ariadne at sales@ariadnesoftware.co.uk.
If you have been issued with a license key, proceed as follows.

On your AS/400, issue the following command to add a license key to enable you to use the Communiqué licensed program:

```
addlickey      lickeyinp(*prompt)
                prdid(1CMNQUE)
                licterm(V3R1)
                feature(5001)
                serial(*local)
                prcgrp(Ppp)
                lickey(XXXXXX YYYYYY ZZZZZZ)
                usglmt(*nomax)
                expdate(exp_date)
                vnddta(*none)
```

where: **XXXXXX YYYYYY ZZZZZZ** is the license key you received by e-mail from us
exp_date is the expiry date mentioned in the e-mail
and: **Ppp** is the processor group mentioned in the e-mail

Communiqué is now ready for you to use!

If you have any trouble at any point in this procedure, please contact us by e-mail at [**support@ariadnesoftware.co.uk**](mailto:support@ariadnesoftware.co.uk).

Maintenance

Periodically we will release Program Temporary Fixes (PTFs) in order to add new features or to resolve any problems found with **Communiqué**.

PTFs are supplied in the form of an AS/400 save file stored within a zip file, and can be downloaded from the Download section of the ariadne web site at

<http://www.ariadnesoftware.co.uk>.

The procedure for loading and applying a PTF for Communiqué is as follows.

First download the PTF you require from the Internet. The PTF will be held in a file called Q1CMnnnn.zip, where nnnn is the PTF id.

This section describes two possible procedures for copying the AS/400 save file held in this zip file to your AS/400. We are hoping to simplify and automate this area in the near future. However, for the moment, please select one of the two methods described below and follow the steps carefully and in the order indicated. Steps to be carried out on the PC are shown in **green**, steps to be carried out on the AS/400 in **blue**.

Method 1: Installing a PTF using FTP

This method uses FTP (File Transfer Protocol) to transfer the PTF file from your PC to your AS/400. It is assumed that you have stored the zip file which you downloaded from the Internet on your PC and that you have enabled an FTP link between your PC and your AS/400.

Step 1

On your PC, create a directory for the PTF installation, either within Windows Explorer, or from the DOS prompt, or re-use the directory you used to install originally, e.g.:

```
md c:\CMNQUEV1R1
```

Save or copy the zip file that you downloaded or received into this directory.

Step 2

On your PC, unzip the contents of the zip file into the directory you just created using pkunzip, WinZip, Norton UnZip etc.

Step 3

On your AS/400, first sign off then sign back on again to ensure that you do to have any Communiqué objects in your QTEMP library. Then decide which library you wish to use for the PTF install procedure and create a save file in that library. The name of this save file should be Q1CMnnnn, where nnnn is the PTF id, e.g.

```
crtsavf lib_name/Q1CM0001
```

where **lib_name** is the name of the library you chose to use.

Step 4

On your PC, start an FTP session to your AS/400. You may use FTP packages such as CuteFTP or simply go to the DOS prompt and type:

```
ftp AS400_name
```

where AS400_name is the name of your AS/400.

You will be prompted to enter your user id and password. Please ensure that you sign on with a user id that has adequate authority. We recommend that you sign on as the Security Officer QSECOFR.

Issue the following FTP commands within the FTP session:

binary

This command instructs FTP not to carry out any conversion on the data to be sent to the AS/400.

cd lib_name

where lib_name is the name of the library in which you created the save file in Step 3. This command makes lib_name your current library on the AS/400.

lcd dir_name

where dir_name is the name of the directory on your PC in which you unzipped the install files earlier. This command makes dir_name your current directory on the PC.

put Q1CMnnnn.sav Q1CMnnnn

This command transmits the contents of the file Q1CM0001.sav, which was earlier unzipped from Q1CM0001.zip, and stores them in the save file called Q1CMnnnn which you created earlier on your AS/400.

Step 5

On your AS/400, ensure that the previous step worked by displaying the contents of save file Q1CMnnnn:

dspsavf lib_name/Q1CMnnnn

If you see the message **(No records in save file)** then something has gone wrong during the preceding steps. Try following them again. If you still cannot get this procedure to work, contact us at support@ariadnesoftware.co.uk.

Step 6

On your AS/400, issue the following command to load the PTF on to your system:

```
lodptf          licpgm(1CMNQUE)
                dev(*savf)
                select(1CMnnnn)
                savf(lib_name/Q1CMnnnn)
```

where lib_name is the name of the library in which you created the save file in Step 3 and nnnn is the PTF id.

Step 7

On your AS/400, issue the following command to apply the PTF to your system:

```
apyptf          licpgm(1CMNQUE)
                select(1CMnnnn)
                apy(*temp)
```

where: **nnnn** is the PTF

If you have any trouble at any point in this procedure, please contact us by e-mail at support@ariadnesoftware.co.uk.

Method 2: Installing a PTF using the IFS

This method uses a Client Access network drive assigned to the IFS of your AS/400 to transfer the PTF from your PC to your AS/400. It is assumed that you have stored the PTF zip file (which you downloaded from the Internet or received via e-mail) on your PC.

Step 1

On your PC, create an install directory for the PTF, either within Windows Explorer, or from the DOS prompt, or use the directory you created for the original installation, e.g.:

```
md c:\PC_dir
```

where PC_dir is the name of the directory you wish to use for this installation.

Save or move the zip file that you downloaded or received into this directory.

Step 2

On your PC, unzip the contents of the zip file into the directory you just created using pkunzip, WinZip, Norton UnZip etc.

Step 3

On your PC, ensure that you have a network drive assigned to a directory in the root file system of your AS/400's IFS. This can be done in Windows Explorer by selecting Tools, Map Network Drive.

Create a directory in the IFS to receive the PTF. This can be done in Windows Explorer by selecting File, New, Folder, or at the DOS prompt as follows:

```
md I:\IFS_dir
```

where **I:** is the network drive assigned to your AS/400 and IFS_dir is the name of the directory you wish to use for this installation.

Step 4

Copy the contents of your PC's PTF directory created in Step 1 above to the IFS directory created in Step 3 above using Windows Explorer, or from the DOS prompt as follows:

```
copy c:\PC_dir\*. * I:\IFS_dir
```

where **PC_dir** is the name of the directory on your PC, **I:** is the network drive assigned to your AS/400 and **IFS_dir** is the name of the directory you wish to use for this installation.

Step 5

On your AS/400, first sign off then sign back on again to ensure that you do to have any Communiqué objects in your QTEMP library. Then decide which library you wish to use for the install procedure and create a save file in that library. The save file should be called **Q1CMnnnn**, where **nnnn** is the PTF id, e.g.

```
crtsave lib_name/Q1CMnnnn
```

where **lib_name** is the name of the library you chose to use

and **nnnn** is the PTF id.

Step 6

On your AS/400, copy the contents of the **Q1CMnnnn.sav** file (where **nnnn** is the PTF id) from the IFS directory to the save file just created, by issuing the following command:

```
cpyfrmstmf      fromstmf('/IFS_dir/Q1CMnnnn.sav')
                tombr('/qsys.lib/lib_name.lib/Q1CMnnnn.file')
                mbropt(*replace)
                cvtdta(*none)
                endlinfmt(*fixed)
                tabexpn(*no)
```

where **IFS_dir** is the name of the IFS directory into which you copied the PTF files in Step 4 above, and **lib_name** is the name of the library in which you created the save file in Step 5 above.

Step 7

On your AS/400, ensure that the previous step worked by displaying the contents of save file **Q1CMnnnn**:

```
dspsavf lib_name/Q1CMnnnn
```

If you see the message (**No records in save file**) then something has gone wrong during the preceding steps. Try following them again. If you still cannot get this procedure to work, contact us at support@ariadnesoftware.co.uk.

Step 8

On your AS/400, issue the following command to load the PTF on to your system:

```
lodptf          licpgm(1CMNQUE)
                dev(*savf)
                select(1CMnnnn)
                savf(lib_name/Q1CMnnnn)
```

where **lib_name** is the name of the library in which you created the save file in Step 3 and **nnnn** is the PTF id.

Step 9

On your AS/400, issue the following command to apply the PTF to your system:

```
apypft          licpgm(1CMNQUE)
                select(1CMnnnn)
                apy(*temp)
```

where: **nnnn** is the PTF

If you have any trouble at any point in this procedure, please contact us by e-mail at support@ariadnesoftware.co.uk.

Using Communiqué

The Communiqué Menu

The easiest way to get started with **Communiqué** is to use **Communiqué** menu. The **Communiqué** menu brings together all of **Communiqué's** command-line options in a single convenient place.

To display the **Communiqué** menu, enter the following command at an AS/400 command line:

GO CMNQUEV1R1/CMNQUE

CMNQUE

Communique Menu

Select one of the following:

1. Send Communique email message

10. Work with address lists
11. Create an address list
12. Change an address list
13. Copy an address list
14. Delete an address list
15. Convert distribution list to address list

20. Work with address list entries
21. Add address list entry
22. Change address list entry
23. Copy address list entries
24. Remove address list entry

Selection or command

===>

F3=Exit F4=Prompt F9=Retrieve F12=Cancel

F13=Information Assistant F16=AS/400 main menu

In the following sections we will explore the various options available from this menu.

Sending a Communiqué Email Message

Option 1 from the ***Communiqué*** menu runs the **SNDCMNMSG** (Send Communiqué Message) command which allows you to send an email message.

The same functionality is available from the ***Communiqué*** API (Application Program Interface), which is described in a later section of this document.

Command parameters are as follows.

TO

The **TO** (Recipients) parameter allows you to specify the email addresses to which the email message should be sent.

You can define up to 32 recipients for the message on this command parameter. If you need to send the same email address to more than 32 recipients simultaneously, you can do this by defining an address list and specifying the address list name on this parameter.

There are 3 elements to this parameter: **Email address**, **Name** and **Type**.

Email address

This is where you enter the email address to which the message is to be sent.

Note that while ***Communiqué*** will check that the email address that you enter conforms to the rules for valid email addresses, it is not possible to validate that the email address that you enter is correct or that the message will be deliverable.

For example, sales.ariadnesoftware.co.uk is not a valid email address (since it does not contain an @ sign), and ***Communiqué*** will reject it. However, sales@ariadnesoftware.org.uk is a valid email address and ***Communiqué*** will allow it, but it is not @riadne's correct email address (it should be sales@ariadnesoftware.co.uk) and the message will not be received.

Name

If you would like your email message to display the recipient's name rather than the email address when it is delivered, enter the name here.

The default value is ***NONE**, i.e. no name is provided and the email address will appear as the recipient instead.

For example, if you specify:

SNDCMNMSG TO((Sales@ariadnesoftware.co.uk *NONE))

when the message is received, the **To:** attribute will be shown as:

To: Sales@ariadnesoftware.co.uk

However, if you specify:

SNDCMNMSG TO((Sales@ariadnesoftware.co.uk '@riadne Sales'))

when the message is received, the **To:** attribute will be shown as:

To: @riadne Sales

Type

Specify the type of recipient here.

Options are:

*PRI	(Default) Primary recipient.
*CC	Carbon copy recipient. An *CC recipient receives a copy of the message, and is identified to the primary recipient, but is not the primary recipient.
*BCC	Blind carbon copy recipient. An *BCC recipient receives a copy of the message, but is not identified to the primary recipient or *CC recipients.
*ADRL	Address list. If you wish to send to an address list, this is the value that must be entered.

Example:

Sending to @riadne Sales as a primary recipient with a copy to @riadne Marketing:

**SNDCMNMSG TO((Sales@ariadnesoftware.co.uk 'Sales' *PRI)
(Marketing@ariadnesoftware.co.uk 'Marketing' *CC))**

Example:

Sending to an email address list called “Sales”:

SNDCMNMSG TO((Sales *ADRL *ADRL))

FROM

The **FROM** (Sender) parameter allows you to specify the sender of the email and the email address to which a response should be sent.

The default value is ***CURRENT**, which means that **Communiq  ** will try to retrieve the email address of the user sending the email from the System Distribution Directory. If no email address can be determined for the user from the System Distribution Directory, you will need to enter the values you wish to use manually.

There are 2 elements to this parameter: **Email address** and **Name**.

Email address

This is where you enter the email address of the sender, which will also be the address to which, by default, a reply to this email should be directed.

Note that while **Communiq  ** will check that the email address that you enter conforms to the rules for valid email addresses, it is not possible to validate that the email address that you enter is correct or that any reply sent to the message will be deliverable.

For example, sales.ariadnesoftware.co.uk is not a valid email address (since it does not contain an @ sign), and **Communiq  ** will reject it. However, sales@ariadnesoftware.org.uk is a valid email address and **Communiq  ** will allow it, but it is not @riadne’s correct email address (it should be sales@ariadnesoftware.co.uk) and any reply sent to this email address will not be received.

If ***CURRENT** is specified, the following method is used to obtain the sender’s email address:

- a) The following fields are selected from the System Distribution Directory entry of the current user (this can be changed using the WRKDIRE command).
 - user id (USRID)
 - user address (USRADDR)
 - SMTP user id (SMTPUSRID)
 - SMTP domain (SMTPDMN)
- b) We then build a sender email address using the following logic:

```
If SMTP user id and domain are non-blank
  Email address = SMTPUserId@SMTPdomain
Else
  If user id, user address, hostname and domain are all non-blank
    Email address = UserId.UserAddress@HostName.Domain
  Else
    If host name and domain are non-blank
```

```

        Email address = UserProfile@HostName.Domain
    Else
        Error CMQ9105 (Unable to identify email address for X).
    EndIf
EndIf
EndIf

```

Name

If you would like your email message to display a sender's name rather than the sender email address when it is delivered, enter the name here.

The default value is ***NONE**, i.e. no name is provided and the email address will appear as the sender instead.

For example, if you specify:

SNDCMNMSG... FROM((Sales@ariadnesoftware.co.uk *NONE))

when the message is received, the **From:** attribute will be shown as:

From: Sales@ariadnesoftware.co.uk

However, if you specify:

SNDCMNMSG FROM((Sales@ariadnesoftware.co.uk '@riadne Sales'))

when the message is received, the **From:** attribute will be shown as:

From: @riadne Sales

If ***CURRENT** is specified, the following method is used to determine the sender's name:

- a) The following fields are selected from the System Distribution Directory entry of the current user (this can be changed using the WRKDIRE command).
 - first name (FSTNAM)
 - last name (LSTNAM)
 - preferred name (PREFNAM)
 - user description (USRD)
 - surname (SURNAM)
 - given name (GIVENNAM)
 - user address (USRADDR)

- b) We then build a sender name using the following logic:

```

If preferred name is non-blank
    Sender name = preferred name
Else
    If first name and last name are non-blank
        Sender name = "FirstName LastName"
    Else
        If given name and surname are non-blank
            Sender name = "GivenName Surname"
        Else
            If user description is non-blank
                Sender name = user description
            Else
                Retrieve user profile attributes
                If text description of user profile is non-blank
                    Sender name = user profile text description
                Else
                    Sender name = blanks (no sender name will be used and the email address
                        will show as the sender)
                End If
            End If
        End If
    End If
End If

```

```
        End If
      End If
    End If
  End If
```

SUBJECT

The **SUBJECT** (Subject) parameter allows you to define a subject line for the message. You can enter up to 50 characters of free-format text.

Example:

```
SNDCMNMSG...SUBJECT('Example subject line')
```

When this message is received, it will show as:

```
Subject:    Example subject line
```

MSG

The **MSG** (Message) parameter allows you to enter the text of an email message directly on the command line.

Up to 512 characters of free-format text can be entered here. If you need to send more than 512 characters of text, then you can either use an embedded text or HTML file (see the **ATTACH** parameter below), or use the **Communiqué** API, which allows up to 4,096 characters of text on the equivalent parameter.

The message can be sent in either plain text, HTML or alternative plain text/HTML formats.

There are 2 elements to this parameter: **Message text** and **Message format**.

Message text:

You can enter the text of a short message here. Up to 512 characters can be entered.

When received, the message will be displayed exactly as it is entered, with the following exceptions:

- If you want to force a line break, enter
. Even if the message is sent in plain text format, this HTML control will be interpreted and converted to a hard line break (carriage return-line feed sequence).
- Other HTML controls may be entered, but will only be interpreted as HTML controls if the message is sent and delivered in HTML format.

Message format:

This is where you specify the format in which the message is sent.

Options are:

*BOTH	(Default) The message is sent in alternative plain text/HTML format. This means that two copies of the message text will be sent: a plain text copy and an HTML copy. If the email client software used to receive the message can handle HTML messages, the HTML copy will be used, otherwise the plain text copy will be used.
*TEXT	The message is sent in plain text format. The only HTML control which is interpreted is <code>
</code> , which Communiq� will convert to a hard line break.
*HTML	The message is sent in HTML format. You can include HTML formatting (e.g. <code> </code> or <code><u> </u></code> to control bold text and underlining). Communiq� will take the text that you enter and wrap it with some basic HTML header and footer controls (<code><HTML> <HEAD> <BODY></code>). These controls should not therefore be included in the text of the message.

Example:

SNDCMNMSG...

**MSG('Here"s a message
with
line
breaks.' *TEXT)**

When this message is received, it will show as:

```
Here's a message
with
line
breaks.
```

Example:

SNDCMNMSG...

**MSG('Here"s a demonstration of a message with HTML controls.

This line is in bold,

<u>While this line is underlined.</u>')**

When this message is received, it will show as:

```
Here's a demonstration of a message with HTML controls.
This line is in bold,
While this line is underlined.
```

ATTACH

The **ATTACH** (Attachments) parameter allows you to attach or embed up to 32 files to your email message.

There are 4 elements to this parameter: **File name, Method, Content type and Code page.**

File name:

This is where you enter the full path name of the file you wish to attach. The path name must be entered in IFS format. In other words, documents stored in shared folders must be referenced in the QDLS file system and database file members must be reference in the QSYS.LIB file system.

Method:

This is where you specify the method by which the file is sent.

Options are:

- | | |
|----------------|--|
| *ATTACH | (Default) The file is sent as an attachment. It will appear as an attached file separate from the text of the email. |
| *EMBED | The contents of the file are embedded in the text of the email and will follow the text of any message entered on the MSG parameter. Please note that your client email software is likely only to support the embedding of certain types of file, e.g. text, HTML and images. |

Content type

The content type tells **Communiq ** and the receiving email software what type of data the attachment represents.

It is important that the content type is specified accurately, as **Communiq ** will process the attachment differently depending on the value you specify here. For example, if you specify the attachment type as *TEXT or *HTML, it will be handled using a technique appropriate to textual data, and may be converted from EBCDIC to ASCII, whereas any other content type will cause the attachment to be handled as binary data and sent unaltered.

Moreover, the receiving software may not handle the attachment appropriately if the content type is specified incorrectly. For example, if you send a PDF file attachment, but specify a content type of HTML, the receiving email software may attempt to display the PDF file as HTML which will cause unwanted results.

Options are:

- | | |
|--------------|---|
| *FILE | (Default) The content type will be inferred from the extension part of the file name. |
|--------------|---|

You can control which file extensions are identified as which content types by listing file extensions in the second-level text of Slipstream messages CMQ5000-CMQ5004. Each file extension must be from 1-5 characters long, padded with trailing blanks if less than 5 characters long. For example, message CMQ5001 controls which file extensions identify image files. This message is supplied with the value:

"JPG JPEG GIF BMP PNG TIF TIFF".

You may change the text of this message, to add or remove extensions, so long as you ensure that each extension is exactly 5 characters long (including any trailing blanks) and remembering that installing PTFs or a new release will overwrite your changes.

Any file extension not listed in one of these messages will be treated as application binary data.

*BINARY	Application binary. This is the default MIME content type used for data types other than those which have distinct content types of their own (image, audio, video etc.)
*TEXT	Identifies the attachment as plain ASCII text.
*HTML	Identifies the attachment as HTML.
*IMAGE	identifies the attachment as an image.
*VIDEO	identifies the attachment as a video file.
*AUDIO	identifies the attachment as an audio file.

Code page

This element allows you to identify the code page of the data in the attachment file. This value is used if the content type is indicated as *TEXT or *HTML to convert the data to ASCII format so it can be displayed by the client email software.

Options are:

*STMF	(Default) The code page attribute of the file in the IFS is used.
1-65534	The code page to use.

Example:

Attaching a PDF file:

SNDCMNMSG...

ATTACH(('/home/sales/salesreport.pdf *ATTACH *BINARY))

Example:

Embedding an HTML file:

SNDCMNMSG...

ATTACH(("home/sales/salesreport.htm *EMBED *HTML))

PRIORITY

When the **SNDCMNMSG** command is prompted using F4, the **PRIORITY** (Priority) parameter only appears if you then also press F10 to display additional parameters.

Options are:

- *NORMAL** (Default) The message is sent specifying normal priority. When the message arrives, the client email software will not mark it as high or low priority.
- *HIGH** High priority. When the message arrives, the client email software will mark it as high priority.
- *LOW** Low priority. When the message arrives, the client email software will mark it as low priority.

CFMDEL

When the **SNDCMNMSG** command is prompted using F4, the **CFMDEL** (Confirm Delivery) parameter only appears if you then also press F10 to display additional parameters.

Options are:

- *NO** (Default) No confirmation of delivery if requested.
- *YES** The message is sent with an indication that you have requested that the recipient return confirmation of delivery. When the message is opened, if you have not switched off this feature, the client software will either send a confirmation message back to the sender of the email or ask you whether you wish to send such a confirmation.

SENDMLTMSG

When the **SNDCMNMSG** command is prompted using F4, the **SENDMLTMSG** (Send Multiple Messages) parameter only appears if you then also press F10 to display additional parameters.

Options are:

- *NO** (Default) If you are sending an email to more than one recipient, a single message will be created with multiple recipients. When the message is received, the reader will be able to see the names of all people to whom the message was sent.
- *YES** If you are sending an email to more than one recipient, multiple separate messages will be sent, one to each recipient. When the message is received, the reader will see only their own name and will not be able to see the names of all people to whom the message was sent.

Working with Address Lists

Option 10 from the **Communiqué** menu runs the **WRKADRL** (Work with Address Lists) command which allows you to create and manage address lists. Address lists provide a means of grouping email addresses so that you can conveniently send messages to groups of people, e.g. departments, customers who have bought a particular product etc.

Selecting option 10 or entering the **WRKADRL** command displays the address lists that you been defined so far.

Address list information is stored in file CM_ADRLST. If you wish to ensure that only a restricted number of people can maintain address lists, you should set the authorities on this file appropriately, for example by associating an authorization list with it.

Work with Address Lists

System: YOURSYS

Type options, press Enter.

1=Create 2=Change 3=Copy 4>Delete 5=Display 6=Print list entries
8=Work with list entries

Opt	Address List	Text 'description'
	FINANCE	Finance Department
	IT	I.T. Department
	MARKETING	Marketing Department
	PRODUCTION	Production Department
	SALES	Sales Department

Bottom

F3=Exit F5=Refresh F12=Cancel F17=Position to

Options that can be input against entries in the list are:

1=Create

Select this option to create a new address list. This option is available only on the first line of the display. You can enter the name of the new

address list in the space available if desired. This is equivalent to running the **CRTADRL** (Create Address List) command.

2=Change

Select this option to change the details of an existing address list. This is equivalent to running the **CHGADRL** (Change Address List) command.

3=Copy

Select this option to copy an existing address list and create a new address list. This is equivalent to running the **CPYADRL** (Copy Address List) command.

4=Delete

Select this option to delete one or more address lists. This is equivalent to running the **DLTADRL** (Delete Address List) command.

5=Display

Selection this option to display the details of an address list.

6=Print list entries

Select this option to print the details of an address this and the entries it contains.

8=Work with list entries

Select this option to delete one or more address lists. This is equivalent to running the **WRKADRLE** (Delete Address List) command.

Available command keys are:

F3=Exit

Exit the application.

F5=Refresh

Refresh the list.

F12=Cancel

Return to the previous screen.

F17=Position

Display the positioner window to allow the list to be positioned to a specified address list.

Creating an address list

Option 11 from the **Communiqué** menu runs the **CRTADRL** (Create Address List) command which allows you to create an address list.

Command parameters are as follows:

ADRL

The **ADRL** (Address list) parameter allows you to specify the name which will be used to identify the address list. The name must be unique and must conform to the normal rules for iSeries object names, except that it may be up to 20 characters in length.

TEXT

The **TEXT** (Text 'Description') parameter enables you to enter up to 50 characters of free-format text to describe and further identify the address list, its contents and purpose.

Example:

Creating an address list to store the email addresses of members of the Sales Department.

```
CRTADRL ADRL(SALES) TEXT('Sales Department')
```

Changing an address list

Option 12 from the **Communiqué** menu runs the **CHGADRL** (Change Address List) command which allows you to change an existing address list.

Command parameters are as follows:

ADRL

The **ADRL** (Address list) identifies the address list you wish to change.

TEXT

The **TEXT** (Text 'Description') parameter enables you to enter up to 50 characters of free-format text to describe and further identify the address list, its contents and purpose.

Example:

Changing the text description on the Sales Department address list.

```
CHGADRL ADRL(SALES) TEXT('Sales Department address list')
```

Copying an address list

Option 13 from the **Communiqué** menu runs the **CPYADRL** (Copy Address List) command which allows you to copy an existing address list and create a new one.

Command parameters are as follows:

FROMADRL

The **FROMADRL** (From address list) identifies the address list you wish to copy.

TOADRL

The **TOADRL** (To address list) identifies the address list you wish to create.

TEXT

The **TEXT** (Text 'Description') parameter enables you to enter up to 50 characters of free-format text to describe and further identify the new address list, its contents and purpose.

CPYADRLE

The **CPYADRLE** (Copy address list entries?) allows you to indicate whether you want to copy the contents of the existing address list or just its description.

Options are:

- *YES** (Default) Yes. The contents of the address list are copied.
- *NO** No. The contents of the address list are not copied.

Example:

Copy the Sales Department address list to create a combined Sales and Marketing list.

```
CPYADRL
  FROMADRL(SALES)
  TOADRL(SALES_AND_MARKETING)
  TEXT('Sales and Marketing address list')
  CPYADRLE(*YES)
```

Deleting an address list

Option 14 from the **Communiqué** menu runs the **DLTADRL** (Delete Address List) command which allows you to delete an existing address list.

Command parameters are as follows:

ADRL

The **ADRL** (Address list) identifies the address list you wish to delete.

Example:

Deleting the Sales Department address list.

```
DLTADRL ADRL(SALES)
```

Converting a Distribution List to an Address List

Option 15 from the **Communiqué** menu runs the **CVTDSTADRL** (Convert Distribution List to Address List) command which allows you to convert a SNADS distribution list to an address list for use with **Communiqué**.

If you were formerly a user of Office Vision/400, you may have SNADS distribution lists which are equivalent to **Communiqué** address lists and which you do not wish to have to recreate. This command allows you to automate the conversion of your existing SNADS distribution lists and create **Communiqué** address lists from them.

Command parameters are as follows:

DSTL

The **DSTL**(Distribution List) parameter identifies the distribution list you wish to convert.

ADRL

The **ADRL** (Address List) parameter specifies the name of the new address list your wish to create.

Options are:

*DSTL	(Default) The name of the address list is derived from the name of the distribution list by concatenating the two parts of the distribution list name with an intervening underscore. For examples, a distribution list called 'ARIADNE SALES' will become an address list called ARIADNE_SALES.
'Address list name'	Enter a valid address list name of your own choosing.

TEXT

The **TEXT** (Text 'Description') parameter enables you to enter up to 50 characters of free-format text to describe and further identify the new address list, its contents and purpose.

HOSTNAME

The TCP/IP host name to be used when converting distribution list entries to email addresses.

Options are:

*SYS	(Default) The currently defined local host name as specified by the CHGTCPDMN (Change TCP/IP Domain) command is retrieved and used.
'Host name'	Enter a valid TCP/IP host name of your own choosing.

DOMAIN

The TCP/IP domain name to be used when converting distribution list entries to email addresses.

Options are:

*SYS	(Default) The currently defined local domain name as specified by the CHGTCPDMN (Change TCP/IP Domain) command is retrieved and used.
'Domain name'	Enter a valid TCP/IP domain name of your own choosing.

Example:

Converting the old SNADS Sales Department distribution list to an address list.

```
CVTDSTADRL
  DSTL(ARIADNE SALES)
  ADRL(*DSTL)
  HOSTNAME(*SYS)
  DOMAIN(*SYS)
```

Communiq  retrieves the contents of the SNADS distribution list and creates a new address list using the value of the **ADRL** parameter.

For each entry in the address list, **Communiq ** then looks at the System Distribution Directory for the user concerned and constructs an email address according to the following rules:

1. If the "Preferred Address type" is set to *SMTP, and an SMTP user id and SMTP domain are defined for the user, these are used and the email address will be in the form: SMTPUserId@SMTPDomain.
2. If the "Preferred Address type" is set to *USRID, the two parts of the user id are concatenated with an intervening period, and these are added to the specified host and domain names to form the email address, e.g. UsrId1.UsrId2@hostname.domainname.
3. If neither of the above rules can be used, the user profile will be added to the host name and domain name specified to give an email address in the form: UserProfile@hostname.domainname.

Communiq  also derives the name of the user concerned according to the following rules:

1. If a "Preferred name" is available, this is used.
2. If a "First name" and "Last name" are available, these are used.
3. If a "Given name" and "Surname" are available, these are used.
4. If a "User description" is available, it is used.
5. If the user profile has text associated with it, it is used.
6. Otherwise the name is blank.

Working with Address List Entries

Address list entry information is stored in file CM_LSTENT. If you wish to ensure that only a restricted number of people can maintain address lists, you should set the authorities on this file appropriately, for example by associating an authorization list with it.

Option 20 from the **Communiqué** menu runs the **WRKADRLE** (Work with Address List Entries) command which allows you to add and manage address list entries.

Selecting option 20 or entering the **WRKADRLE** command allows you to select an address list and display the address list entries that you have defined so far.

There is a single command parameter

ADRL

The **ADRL** (Address list) identifies the address list whose entries you wish to work with.

Alternatively, use the **WRKADRL** (Work with address lists) command and enter option 8=Work with address list entries against the address list in which you are interested.

```
-----  
                                Work with Address List Entries  
                                System:    YOURSYS  
Address list . . . . . :    SALES  
  
Text 'description' . . :    Sales Department  
  
Type options, press Enter.  
  1=Add   2=Change   4=Remove   5=Display   6=Print  
  
Opt      Email address  
  
        Jane.Doe@ariadnesoftware.co.uk  
        Joe.Bloggs@ariadnesoftware.co.uk  
        John.Smith@ariadnesoftware.co.uk  
  
                                Bottom  
F3=Exit   F5=Refresh   F12=Cancel   F17=Position to  
-----
```

Options that can be input against entries in the list are:

- | | |
|------------------|--|
| 1=Add | Select this option to add a new address list entry. This option is available only on the first line of the display. You can enter the new email address in the space available if desired. This is equivalent to running the ADDADRLE (Add Address List Entry) command. |
| 2=Change | Select this option to change an existing address list entry. This is equivalent to running the CHGADRLE (Change Address List Entry) command. |
| 4=Remove | Select this option to remove one or more address list entries. This is equivalent to running the RMVADRLE (Remove Address List Entry) command. |
| 5=Display | Select this option to display the details of an address list entry. |
| 6=Print | Select this option to print an address list entry. |

Available command keys are:

- | | |
|---------------------|---|
| F3=Exit | Exit the application. |
| F5=Refresh | Refresh the list. |
| F12=Cancel | Return to the previous screen. |
| F17=Position | Display the positioner window to allow the list to be positioned to a specified address list. |

Adding an address list entry

Option 21 from the **Communiqué** menu runs the **ADDADRLE** (Add Address List Entry) command which allows you to add an entry to an address list.

Command parameters are as follows:

ADRL

The **ADRL** (Address list) parameter allows you to specify the name of the address list to which the entry is to be added.

EMAIL

The **EMAIL** (Email address) parameter allows you to specify the email address to be added to the list. This must be a valid (i.e. well formed) email address and must be unique within the address list.

NAME

The **NAME** (Name) parameter enables you to enter the name of the person or organization the email address relates to.

TYPE

The **TYPE** (Type) parameter defines the manner in which messages are sent to the email address being added to the list when the address list is used.

Options are:

*PRI	(Default) Primary recipient.
*CC	Carbon copy recipient. An *CC recipient receives a copy of the message, and is identified to the primary recipient, but is not the primary recipient.
*BCC	Blind carbon copy recipient. An *BCC recipient receives a copy of the message, but is not identified to the primary recipient or *CC recipients.

Example:

Adding an entry to the Sales Department address list.

```
ADDADRLE
  ADRL(SALES)
  EMAIL('Joe.Bloggs@ariadnesoftware.co.uk')
  NAME('Joe Bloggs')
  TYPE(*PRI)
```

Changing an address list entry

Option 22 from the **Communiqué** menu runs the **CHGADRLE** (Change Address List Entry) command which allows you to change an address list entry.

Command parameters are as follows:

ADRL

The **ADRL** (Address list) parameter identifies the address list in which the entry you wish to change can be found.

EMAIL

The **EMAIL** (Email address) parameter identifies the address list entry you wish to change.

NAME

The **NAME** (Name) parameter enables you to enter the name of the person or organization the email address relates to.

TYPE

The **TYPE** (Type) parameter defines the manner in which messages are sent to the email address being added to the list when the address list is used.

Options are:

*PRI	(Default) Primary recipient.
*CC	Carbon copy recipient. An *CC recipient receives a copy of the message, and is identified to the primary recipient, but is not the primary recipient.
*BCC	Blind carbon copy recipient. An *BCC recipient receives a copy of the message, but is not identified to the primary recipient or *CC recipients.

Example:

Changing an entry in the Sales Department address list.

```
CHGADRLE  
ADRL(SALES)  
EMAIL('Joe.Bloggs@ariadnesoftware.co.uk')  
NAME('Joseph Bloggs')  
TYPE(*CC)
```

Copying address list entries

Option 23 from the **Communiqué** menu runs the **CPYADRLE** (Copy Address List Entries) command which allows you to copy the contents of an existing address list to another address list.

Command parameters are as follows:

FROMADR

The **FROMADR** (From address list) identifies the address list whose contents you wish to copy.

TOADR

The **TOADR** (To address list) identifies the address list to which you wish to copy the address list entries.

Example:

Copy the contents of the Sales Department address list to a new combined Sales and Marketing list.

```
CPYADRLE
  FROMADR(SALES)
  TOADR(SALES_AND_MARKETING)
```

Removing an address list entry

Option 24 from the **Communiqué** menu runs the **RMVADRLE** (Remove Address List Entry) command which allows you to remove an entry from an address list.

Command parameters are as follows:

ADRL

The **ADRL** (Address list) parameter identifies the address list in which the entry you wish to remove can be found.

EMAIL

The **EMAIL** (Email address) parameter identifies the address list entry you wish to remove.

Example:

Removing Joe Bloggs from the Sales Department address list.

```
RMVADRLE
  ADRL(SALES)
  EMAIL('Joe.Bloggs@ariadnesoftware.co.uk')
```

Using the Communiqué API CM_SNDAPIR

As an alternative to using the **SNDCMNMSG** command, you can also call the **Communiqué** API to send an email message. If you want to send an email from a program written in a language such as RPG or COBOL, this may be easier than using **SNDCMNMSG**. In addition, the length of the free-format message text that can be defined when the API is employed is 4,096 bytes as opposed to 512 bytes when **SNDCMNMSG** is used.

When **Communiqué** is installed, a source file call **CM_SRCFILE** is included in library **CMNQUEV1R1**. This contains two members:

1. An ILE RPG copybook called **CM_SNDAPIR** which contains the definitions you need to call the **CM_SNDAPIR** API.
2. A program called **EXAMPLE** which gives an example of how to call the API.

Communiqué Send Email Message (CM_SNDAPIR) API

Required Parameter Group:

1	Format of sender structure	Input	Char(8)
2	Sender	Input	Char(*)
3	Number of recipients	Input	Binary(4)
4	Format of recipient structures	Input	Char(*)
5	Recipient structures	Input	Array 1-32 of Char(*)
6	Subject	Input	Char(*)
7	Format of message structure	Input	Char(8)
8	Message structure	Input	Char(*)
9	Number of attachments	Input	Binary(4)
10	Format of attachment structure	Input	Char(8)
11	Attachment structures	Input	Array 1-32 of Char(*)
12	Priority	Input	Char(1)
13	Confirm delivery	Input	Char(1)
14	Email message identifier	Output	Char(30)
15	Error Code	I/O	Char(*)

Required Parameter Group

Note that many of the structures shown below include text fields preceded by a 2-byte length. In languages such as ILE RPG which support variable length character fields, these fields can be defined as variable length. For example, the Subject parameter shown as a structure below can be defined in ILE RPG as 50A VARYING.

Format of Sender Structure

INPUT; CHAR(8)

The format of the Sender parameter.

Currently only one value is permitted: *CM_F0100*

Sender Structure

INPUT; CHAR(*)

A structure identifying the sender of the message.

Offset		Type	Field
Dec	Hex		
0	0	BINARY(2)	Length of sender email address
2	2	CHAR(128)	Sender email address
130	82	BINARY(2)	Length of sender name
134	86	CHAR(128)	Sender name

Number of recipients

INPUT; BINARY(4)

The number of recipient structures that follow.

Format of Recipient Structures

INPUT; CHAR(8)

The format of the Recipient structures that follow.

Currently only one value is permitted: *CM_T0100*

Recipient Structures

INPUT; ARRAY 1-32 of CHAR(*)

An array of from 1 to 32 structures identifying recipients of the message.

Each element in the array is a structure in the following format.

Offset		Type	Field
Dec	Hex		
0	0	BINARY(2)	Length of recipient email address
2	2	CHAR(128)	Recipient email address
130	82	BINARY(2)	Length of recipient name
134	86	CHAR(128)	Recipient name or *ADRL
262	106	CHAR(10)	Recipient type: *PRI, *CC, *BCC or *ADRL

Subject

INPUT; CHAR(*)

A structure defining the subject of the message

Offset		Type	Field
Dec	Hex		
0	0	BINARY(2)	Length of subject text
2	2	CHAR(50)	Subject text

Format of Message Structure

INPUT; CHAR(8)

The format of the Message structure that follows.

Currently only one value is permitted: *CM_M0100*

Message Structure

INPUT; CHAR(*)

A structure defining the message to be sent, if any, in the following format.

Offset		Type	Field
Dec	Hex		
0	0	BINARY(2)	Length of message text
2	2	CHAR(4096)	Message text
4098	1002	CHAR(5)	Message type: *BOTH, *TEXT or *HTML

Number of attachments

INPUT; BINARY(4)

The number of attachment structures that follow.

Format of Attachment Structures

INPUT; CHAR(8)

The format of the Attachment structures that follow.

Currently only one value is permitted: *CM_A0100*

Attachment Structures

INPUT; ARRAY 1-32 of CHAR(*)

An array of from 1 to 32 structures identifying attachments to the message.

Each element in the array is a structure in the following format.

Offset		Type	Field
Dec	Hex		
0	0	BINARY(2)	Length of attachment path name
2	2	CHAR(512)	Attachment path name
514	202	CHAR(7)	Attachment method: *ATTACH or *EMBED
521	209	CHAR(7)	Content type: *FILE, *TEXT, *HTML, *IMAGE, *VIDEO, *AUDIO, *BINARY
528	210	BINARY(4)	Attachment code page: -1 = *STMF

Priority

INPUT; CHAR(1)

A code indicating the message priority:

N Normal priority

H High priority

L Low priority

Confirm delivery

INPUT; CHAR(1)

Whether confirmation of delivery is requested:

Y Yes

N No

Email message id

OUTPUT; CHAR(30)

The unique message identifier of the message that was sent, if the call to the API was successful.

This value is the key to the **Communiqué** message log file **CM_MSGLOG**.

Error code

INPUT-OUTPUT; CHAR(*)

A standard iSeries API error structure in the following format.

Offset		Use	Type	Field
Dec	Hex			
0	0	INPUT	BINARY(4)	Bytes provided
4	4	OUTPUT	BINARY(4)	Bytes available
8	8	OUTPUT	CHAR(7)	Exception ID
15	F	OUTPUT	CHAR(1)	Reserved
16	10	OUTPUT	CHAR(*)	Exception data

The Communiqué Database

Communiqué includes a number of database tables which you may find useful. These are listed and explained below.

Message Log CM_MSGLOG

Message Log file CM_MSGLOG logs each email message that is sent.

Each message is identified by a 30-byte message identifier. This is the message identifier which is displayed in the completion message sent by the **SNDCMNMSG** command (“Communiqué message XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX sent successfully.” and returned by the **CM_SNDAPIR** API.

Each log record also contains details:

- when the message was sent
- by which job it was sent
- by which user it was sent
- sender email address
- sender name
- subject
- message text
- message type
- priority
- confirm delivery option

Message Recipients CM_MSGRCP

Message Recipients file CM_MSGRCP logs recipients for each email message that is sent.

Each record is identified by the email identifier which is also the key to CM_MSGLOG.

Each record also contains:

- recipient email address
- recipient name
- recipient type

Message Attachments CM_MSGATC

Message Attachments file CM_MSGATC logs attachments for each email message that is sent.

Each record is identified by the email identifier which is also the key to CM_MSGLOG.

Each record also contains:

- attachment file name
- attachment method

- attachment content type
- attachment code page