



OfficePath

Version 1 Release 6.1

Programming Interfaces

OfficePath Programming Interfaces

Copyright

© Copyright 1995-2021 by TBS Software Inc.
All Rights Reserved.

The software described in this publication is furnished under license and may be used or copied only in accordance with the terms of the license agreement.

Trademarks

Product names may be trademarks or registered trademarks of their respective owners. OS/390® and z/OS® are registered trademarks of IBM Corporation. Other company, product, and service names may be trademarks or service marks of others.

Edition Information

September 2021

This edition applies to Version 1 Release 6.1 of OfficePath. It also applies to subsequent releases until otherwise indicated in new editions.

Inquiries may be directed to:

TBS Software Inc.

2201-278 Bloor St E.
Toronto, ON
Canada
M4W 3M4

Phone: 905 940-9373

Fax: 905 940-9376

Web: www.tbssoft.com

eMail: support@tbssoft.com

Page Break

Table of Contents

Preface.....	Error! Bookmark not defined.
Table of Contents.....	3
List of Illustrations.....	8
Chapter 1. Introduction.....	9
Purpose of this book.....	9
Audience for this book.....	9
How this book is organized.....	9
Other books you may need.....	10
Chapter 2. Product Overview.....	11
What is OfficePath?.....	11
Services not provided by OfficePath.....	12
System Overview.....	13
PART 1 - DESIGNING AN APPLICATION.....	15
Chapter 3. Identifying the application requirements.....	16
Batch or CICS?.....	16
Sending, Receiving or Both?.....	16
What types of document will be processed?.....	17
Is status information important?.....	18
Chapter 4. Defining the application to OfficePath.....	19
Defining the input file.....	20
Defining the application user(s).....	20
Defining the output file.....	20
Document types and Character set.....	20
Accumulate Recipients.....	20
Returns status information.....	21
Chapter 5. Creating an application to send mail.....	22
Chapter 6. Obtaining status information from OfficePath.....	24
How to request that status information is returned.....	24
Format of the returned status information.....	26
Chapter 7. Creating an application to receive mail.....	28
Format of the output document.....	28
Chapter 8. Returning status information to OfficePath.....	31
PART 2 - BATCH INTERFACES.....	33

Chapter 9. Sending requests using the batch interface	34
How the interface operates.....	34
Defining a batch application	35
Creating an Intray file	35
The INTRAY record.....	36
Using the interface program.....	37
Input Parameters	38
Document dataset.....	38
Return codes generated.....	39
Sending multiple documents.....	39
Further considerations.....	39
Example application.....	39
Chapter 10. Automating the distribution of Pre-printed Forms	41
Creating the files	41
AUTOERR file	41
PIPANICK file.....	42
Using the AUTOSEND JCL.....	43
AUTOCNTL Control Parameters	44
AUTOSEND Messages	46
Chapter 11. Outbound - using a VSAM file	50
How the interface operates.....	50
Defining an Outbound destination	51
Creating an outtray file	51
Destination record, OUTTRAY type.....	52
User record.....	56
JCL record.....	57
SYSTEM record.....	57
Using the interface	58
Execution Parameters.....	58
Output Parameters.....	59
Output Document.....	59
Return codes generated.....	59
Further considerations.....	60
Example application.....	60
Chapter 12. Outbound - using an Internal Reader	61
How the interface operates.....	61
Defining an Internal reader to OfficePath.....	61
Internal reader definition.....	62
Destination record, Internal reader type.....	63
User record.....	66
The JCL records	67
The Standard OfficePath Users.....	70
Further considerations.....	73

Sample Application.....	73
PART 3 - TSO INTERFACES	74
Chapter 13. Using the TSO interface	75
How the TSO Interface operates.....	75
Sending from TSO	75
Sending to TSO.....	76
Obtaining status information.....	76
Defining the TSO interface to OfficePath	76
Installing the dialogs in TSO	77
The TSO dialogs	78
PIPMENU	78
PIPWORK.....	79
PIPSDOC	79
PIPSMSG.....	81
PIPCDFS.....	81
PIPSEND	82
PART 4 - CICS INTERFACES	83
Chapter 14. Inbound - using the CICS interface program PIPGS3	84
How the interface operates.....	84
Defining a CICS application.....	85
Creating the intray file	85
The INTRAY record.....	86
Using the interface	87
Input Parameters	88
Document.....	89
PIPGS3 Return codes.....	89
Further considerations.....	92
Avoiding processing delays	92
Error recovery	92
Sample CICS inbound application.....	92
Chapter 15. Outbound - using the CICS interface program PIPGT3	95
How the interface operates.....	95
Defining a CICS Outtray	95
Creating an Outtray file	95
Destination record, OUTTRAY type.....	97
User record.....	100
Using The Interface.....	101
Output Parameters.....	102
Document contents.....	103
PIPGT3 Return codes	104
Further Considerations.....	104
START data passed to application.....	104

Error Recovery	105
Sample CICS outbound application.....	105
CICS table entries required.....	107
Appendix.....	108
Appendix A. Parameters used for requests into OfficePath	109
A.1 Input commands.....	109
SEND commands.....	109
ACK commands.....	109
A.2 Parameter format.....	110
A.3 Input parameters.....	111
A.4 Commands and parameters	117
The SEND command	117
The Acknowledge (ACK) command	118
A.5 Example commands	119
SEND MESSAGE command.....	119
SEND NOTE command.....	119
SEND DOCUMENT command.....	120
SEND PROFILE command	120
ACK DELIVERED command.....	121
ACK INVALID command.....	121
ACK CANCELLED command.....	121
ACK ERROR command.....	122
ACK ACTIONED command.....	122
Appendix B. Parameters used for requests out of OfficePath	123
B.1 Output types	123
Received Mail	123
Received Status.....	123
B.2 Parameter format	124
B.3 Output parameters	125
B.4 Output types and parameters.....	130
B.5 Handling a confirmation document.....	132
B.6 Example output types.....	133
Message.....	133
Note.....	133
Document.....	134
Profile.....	134
Confirmation of delivery.....	135
Receipt acknowledgment.....	136
Appendix C. Document type and Character Set Code Numbers	137
Character Set (GCID) codes	137
Document type (DCA) codes.....	137

Index 139

List of Illustrations

1. [OfficePath schematic arrangement](#)
2. [Application to send/receive mail](#)
3. [Request to send a document](#)
4. [Sending a document and requesting status information](#)
5. [Example of a status information document](#)
6. [Example of an output request](#)
7. [Example of how to return a status](#)
8. [The JCL to define an INTRAY file](#)
9. [Sample In tray FCT definition](#)
10. [The INTRAY maintenance panel](#)
11. [Example JCL for running PIPGS3B](#)
12. [JCL to create PIPANICK file](#)
13. [Job steps required to execute the AUTOSEND program](#)
14. [The JCL to define an Out tray file](#)
15. [Sample FCT entry for an Out tray file](#)
16. [The Destination panel for OUTTRAY type destinations](#)
17. [The User maintenance panel](#)
18. [The JCL maintenance panel](#)
19. [Example JCL for running PIPGT3B](#)
20. [Sample JCL for running PIPGT3B](#)
21. [DCT table updates for an Internal reader](#)
22. [The Destination panel for INTRDR type destinations](#)
23. [The User maintenance panel](#)
24. [Example JCL record for TSO user](#)
25. [The PIPMENU panel](#)
26. [The PIPSDOC panel](#)
27. [The PIPMSG panel](#)
28. [The PIPCDFS Panel](#)
29. [The JCL to define an INTRAY file](#)
30. [Sample In tray FCT definition](#)
31. [The INTRAY maintenance panel](#)
32. [Example LINK for PIPGS3](#)
33. [Example panel from the PIPT transaction](#)
34. [Example PIPTEST Destination definition](#)
35. [The JCL to define an Out tray file](#)
36. [SAMPLE FCT definition for an out tray](#)
37. [The Destination panel for OUTTRAY type destinations](#)
38. [The User maintenance panel](#)
39. [Example LINK for PIPGT3](#)
40. [Example panel from the PIPU transaction \(Mail Item\)](#)
41. [Example panel from the PIPU transaction \(COD\)](#)

Chapter 1. Introduction

Purpose of this book

This book provides the information required to develop and install an interface between an OfficePath system and an application based in CICS, MVS/BATCH or TSO. A number of example interfaces are detailed which the reader can either customize or use to create their own interface.

Note: This book describes the interfaces for the Batch-TSO and CICS features of OfficePath. Before developing an application you should ensure that you are licensed to use the appropriate OfficePath feature.

Audience for this book

This book is intended for the:

- Applications developer responsible for developing mail interfaces between OfficePath and an application.
 - Systems programmer responsible for the installation of interfaces between OfficePath and applications.
-

How this book is organized

This book is arranged into four sections:

Design This part considers the design of an application. This is relevant to both Batch and CICS applications.

Batch This part deals with the OfficePath batch interface programs and the internal reader method of distributing to batch. It is for use with the Batch-TSO feature of OfficePath.

TSO This part deals with the CLISTs and panels provided by OfficePath to communicate with the TSO environment. It is for use with the Batch-TSO feature of OfficePath.

CICS This part deals with the OfficePath CICS interface programs. It is for use with the CICS feature of OfficePath.

This section gives a self-contained description of the functions available and the procedure to be followed when setting up each connection. Further explanation of the parameters that can be passed into and out of OfficePath is given in the appendices.

[Other books you may need](#)

There are four volumes that accompany the OfficePath system software:

- *OfficePath Installation and Administration* manual, which explains how to install, customize and administer the OfficePath system.
- *OfficePath Reference and Diagnosis* manual, which describes the detailed working of OfficePath, and also gives a list of the most common problems encountered.
- *OfficePath Programming Interfaces* manual, which provides information on how to develop and install an interface between an OfficePath system and an application based in CICS, MVS/BATCH or TSO.
This manual is only supplied if the CICS or Batch features have been purchased.
- *OfficePath Library Retrieval Feature Installation and User Guide*, which provides information on how to install and use the Library Retrieval feature, providing on-line search and retrieval of documents from an existing DISOSS library.
This manual is only supplied if the Library Retrieval feature has been purchased.

In addition, the following IBM manuals may be required:

- *DisplayWrite/370 Printing Guide* (SH12-5659), which describes how to configure DisplayWrite/370 in order to print documents.
 - *DisplayWrite/370 Version 2 Programming Interfaces for customers* (SH12-5660), which describes the commarea used to interface to DisplayWrite/370.
 - *DISOSS Interchange Architecture Reference Guide* (SH30-3276), which describes the data formats used for by the SNADS interface and also for communicating with Personal Services.
 - *Installing OV/MVS Release 2.0* (SH21-0515), which describes how to customize Personal Services.
-

Chapter 2. Product Overview

What is OfficePath?

OfficePath is a CICS based node in the Office Systems Network. It provides "store and forward" Document Distribution Service facilities for routing from a **source** to one or more **destinations**.

The **source** can be one of the following:

Personal Services	This can be either PS/CICS 1.3 or the Personal Services component of OfficeVision/MVS.
An API application	An API application which uses the same subset of DISOSS commands as Personal Services ¹ .
A SNADS node	Any other SNADS node. This may be another OfficePath system.
A CICS Application	The Mailsack VSAM KSDS is used as an in-tray between the CICS application and OfficePath.
A Batch Application	The Mailbox VSAM KSDS is used as an in-tray between the batch application and OfficePath.
TSO	A TSO User can use the supplied CLISTs to send TSO datasets and/or messages.

The **destination** can be one of the following types:

Personal Services	This can be either PS/CICS 1.3 or the Personal Services component of OfficeVision/MVS.
An API application	An API application which uses the same subset of DISOSS commands as Personal Services ¹ .
A SNADS node	Any other SNADS node. This may be another OfficePath system.
A CICS Application	A VSAM KSDS, called an out-tray, is used as the interface between the CICS application and OfficePath.
An Internal Reader	The document text is included, as in-stream data, in a JCL deck

¹ The OfficePath programs intercept DISOSS API requests by replacing the API module DSVAW100 (or DMDAW130 module for Personal Services). This module is invoked via an EXEC CICS LINK command which contains a list of storage addresses. This means that OfficePath **must** be in the same region as the API programs.

submitted to an internal reader.

A Batch Application The document text is written to a VSAM KSDS, called an out-tray, and a batch job is submitted to process it.

TSO The document is sent to the TSO user using an internal reader. The user can view the document using the supplied CLISTS.

Services not provided by OfficePath

The following facilities are not provided by OfficePath:

1. Document Library Services

The FILE, SEARCH, RETRIEVE and DELETE commands, which may be issued from within a DISOSS API application (for example, Personal Services), are not supported. These requests will be rejected, with an error status message.

2. Password Validation

OfficePath does not support password validation. It is the responsibility of CICS applications to carry out Password validation. OfficePath includes a sample user exit, which is invoked by the batch programs, and may be used to develop a password validation mechanism.

3. RGN (Routing Group Name)

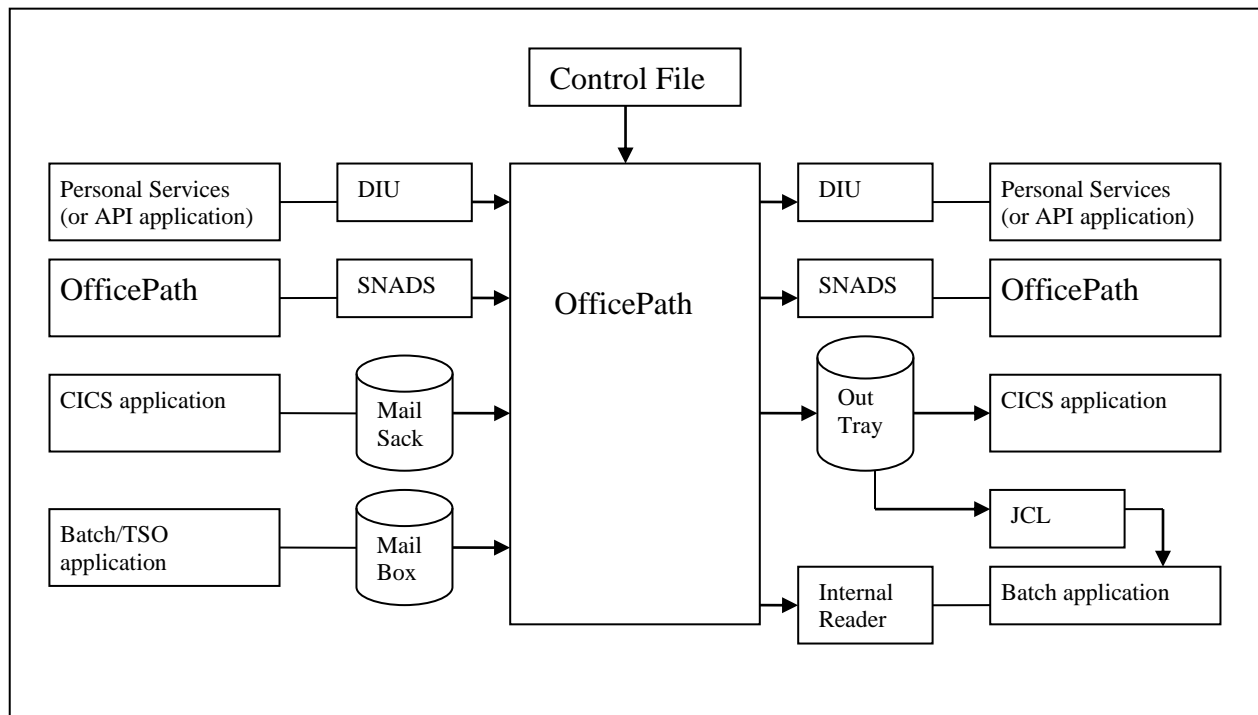
OfficePath does not support the RGN: it is left blank for requests sent to another node and is ignored for requests from other nodes.

Note: The OfficePath Library Retrieval Feature provides on-line search and retrieval of documents from an existing DISOSS library. Full details are in the *OfficePath Library Retrieval Feature Installation and User Guide*.

System Overview

OfficePath is represented schematically as follows:

Figure 1. OfficePath schematic arrangement



In summary, OfficePath provides the following:

- Connectivity to Personal Services or API application
OfficePath communicates with PS in the same way as DISOSS, using IBM's Document Interchange Architecture (DIA). However, OfficePath only supports a limited subset of DISOSS functions, specifically:
 - SIGNON** All DCA levels and GCIDs are ignored. The values from the OfficePath control file are used.
 - SIGNOFF** No options can be specified.
 - REQDIST** All options not involving library documents.
 - LIST** Unformatted LIST command only.
 - OBTAIN** Obtain all documents only.
- The following features are **not** supported:

USERSETS OfficePath does not support USERSETS. However, OfficePath provides an equivalent facility by allowing users to be registered generically (for example all users at the same address can be registered with a single registration record).

FORMAT41 Format 41 addressing in either originator or recipient fields is not supported.

- Connectivity to other OfficePath systems.

OfficePath communicates with other OfficePath systems using IBM's SNA Distribution Services architecture (SNADS).

- Connectivity to CICS, batch and TSO applications.

OfficePath communicates with your applications using proprietary mechanisms which are described in the *OfficePath Programming Interfaces* manual.

- Conversion between the following document types defined by IBM's Document Content Architecture:
 - Revisable Form Text (RFT)
 - Final Form Text (FFT)
 - EDITABLE Notes and Documents

Document conversion is performed by "formatter" programs. For RFT documents, **DisplayWrite/370** can be used to perform conversions; alternatively OfficePath provides sample programs to perform the conversions.

PART 1 - DESIGNING AN APPLICATION

Chapter 3. Identifying the application requirements

This section of the manual gives guidelines on the design decisions which need to be made when writing an application to interface to OfficePath.

The examples shown in this section are valid for both the Batch and CICS interfaces. The only difference is that a Batch application will interface to OfficePath using sequential files, whereas the CICS interface will use Temporary Storage queues.

Batch or CICS?

Generally, you will be using OfficePath to connect an existing application to the IBM Office System. In this case the existing application will determine whether the CICS or Batch interface should be used. If the application runs in Batch then use the Batch interface; otherwise use the CICS interface.

Note: If you are designing a new application then, use the CICS interface if possible because of the sophisticated recovery and data integrity procedures which CICS offers.

Sending, Receiving or Both?

An application may be designed to send requests, receive requests, or both. The simplest of these is an application which needs to send requests through OfficePath without requesting any status information. In this case you need to change your application so that it generates a request, on a sequential file or TSQ, and invokes the OfficePath interface program. Having made the request the application can then continue processing as normal.

If the application is intended to receive mail then it will need to be changed so that it invokes the OfficePath interface program and reacts according to the information returned. Each time the OfficePath interface is called one request will be returned, on either a sequential file or a TSQ. The application should loop around until all requests have been processed. The application can either invoke itself on a timer, to check for new mail, or it can be invoked by OfficePath whenever new mail arrives.

What types of document will be processed?

The main function of OfficePath is to move data (for example, documents or spreadsheets) from one office application to another, for example from Personal Services to TSO. Both the sending and receiving applications need to understand what the data is in order to be able to process it. To achieve this IBM have defined two fields in the DIA architecture; "DCA LEVEL" and "GCID".

The "DCA LEVEL" is a number which specifies the format of the data so that an application knows how to process the document contents.

The main document formats which are supported by the IBM Office system are:

EDITABLE The simplest document format. Each record represents a line of text preceded by an ANSI defined printer control (Similar to standard MVS SYSOUT datasets with RECFM=A).

FF TEXT A data stream in which the text is interspersed with control characters (for example x"06" new-line). The data stream is processed sequentially and each control is processed as it is read.

RF TEXT A data stream where the text is imbedded into text units, preceded by length fields, to enable easy editing.

PC FILE A "generic" DCA LEVEL used to identify "files" created on a PC. The file contents could be anything from a spreadsheet to an executable file.

The "GCID" (or character set) is two numbers which determine how the characters in the document should be interpreted. For example, without GCIDs a document containing the x"5B" character will be interpreted as a pound sterling sign in the UK and as a dollar sign in the US.

690-256	Uppercase Multilingual
337-256	Multilingual
313-285	English UK
101-037	English US

Is status information important?

When mail is sent through OfficePath, status information is available which indicates whether or not the mail arrived successfully. If an application sends mail through OfficePath it can request confirmation that the recipients have received (or actioned) the mail. Alternately, an application which receives mail can return a confirmation to the originator that the mail was received correctly.

Status information can be used by an application to:

- Give positive confirmation that documents have arrived

Applications which are sending requests into OfficePath can request information on the validity of the recipient, so that the originator can be informed. This is important for applications, such as interfaces to mail systems, where the recipient is specified by the originator for each "send" request. If the recipient is generated automatically by the application, it will often be safe to assume that the recipient is valid, and dispense with status processing. Two levels of status can be returned:

DELIVERY In this case a status will be returned when the mail has either been placed in the recipient's in-basket or the recipient is found to be invalid.

**RECEIPT
ACK** In the case a status will be returned when the user first actions the item in his in-basket.

Note: Status information is returned asynchronously, depending on when the recipient processes his mail. Thus a single distribution to several recipients may result in several confirmations being returned, in no set order, over a period of time. This adds complexity to your application design, and therefore you should consider carefully before choosing this feature.

- Inform the originator whether the request was processed or not

When dealing with applications which receive requests from OfficePath, two approaches are possible:

- OfficePath sets the request status to "complete" when it passes the request to the application.

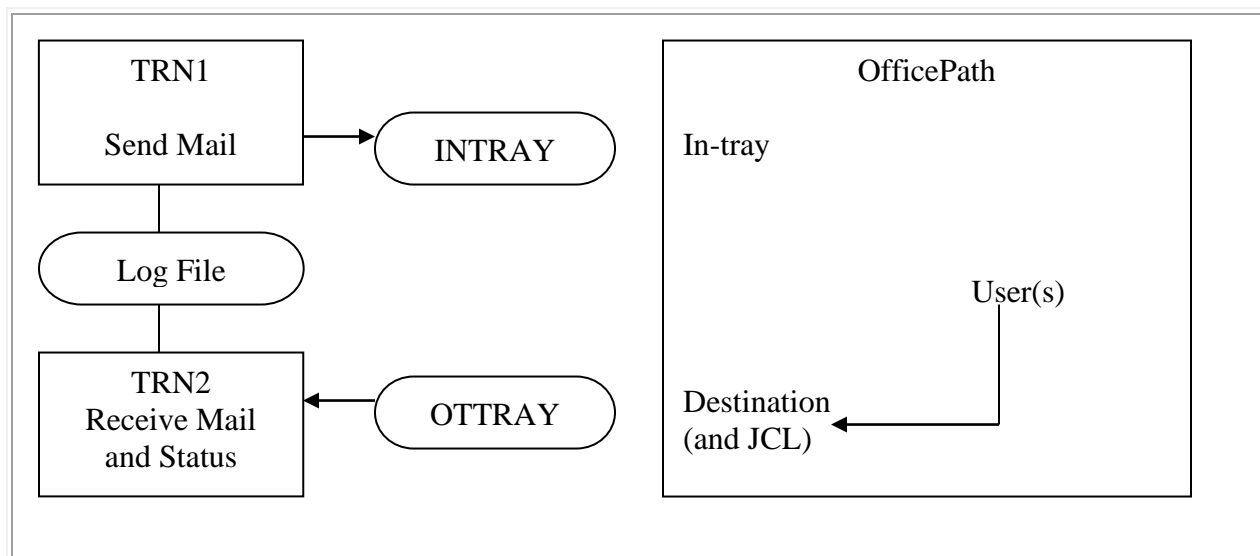
Or

- The application returns status information when it has processed the request.

Chapter 4. Defining the application to OfficePath

OfficePath conforms to the architectures used by DISOSS for distributing mail. This means that mail and status are distributed asynchronously to the recipients. [Figure 2](#) shows a sample application which can send mail and receive status information.

Figure 2. Application to send/receive mail



The following records are required on the OfficePath control file:

- INTRAY** Defines the input file to which the application writes requests.
- USER** Defines the user(s) which the application processes. This record points to the destination record...
- DESTINATION** Defines the characteristics of the application. For example, the out-tray name and the types of document which can be processed.
- JCL** JCL to invoke a batch program. This is required only for batch applications.

If your application is going to send mail and request confirmation then you will require a log-file. This file is used to track requests which have been sent into OfficePath by the TRN1 transaction. When the TRN2 transaction receives status information it can then use the log-file to obtain information about the original request.

Defining the input file

OfficePath requires a record which identifies the input file used by the application. The record is used by OfficePath to determine whether the file is used by a CICS or a batch application. If the file is written by a batch application then OfficePath will issue an MVS ENQ before accessing the file.

Defining the application user(s)

In order to receive mail or status information the recipient must be defined on the OfficePath control file. The user records on the file point to a destination record which determines how the mail is processed for that user. Recipients may be explicitly (for example FRED at WITTON) or generically (for example any userid at LONDON) defined.

Defining the output file

The destination record is used by OfficePath to determine the name of the output file to which the mail/status is written and the transaction which is started to process the mail/status. There are several flags which affect the way which mail is processed, the main flags are described below.

Document types and Character set

You can specify which DCA LEVELs and GCID your application can process. If a document is sent to you in a format which your application cannot process then OfficePath will attempt to translate it, using a transformation module, into a format which your application can process (for example RF Text to EDITABLE). If the document cannot be converted a status is returned to the sender informing him that the recipient could not receive the mail. In addition, the "MESSAGE" document type can be used to prevent users from sending messages (or external documents) to an application which is expecting only documents. If your application does use one of the standard IBM document types then you will need to write transformation modules to convert to and from your document type to one of the IBM document types. This is described in the "Document Conversion" section of the *Installation and Administration Guide*.

Accumulate Recipients

This flag can be used to cause OfficePath to collect together all recipients of one distribution to one application. Normally if a document is sent to several recipients it will be sent through OfficePath and will arrive, at the application, as several documents each with one recipient. If the application is, for example, forwarding these documents into another mailing system this can be inefficient. In this case OfficePath can be instructed, by setting the "accumulate recipients" flag, to collect all recipients for this application and send the document once only, with a list of

recipients. Obviously, the application must be coded to expect to receive more than one recipient in a request, or the extra recipients will be lost. Note: Since each output request can have a maximum of 50 recipients if a request is sent to more than 50 recipients, on one destination, multiple request records will be written.

Returns status information

This flag is used to determine whether OfficePath or your application is going to return status information to the originator of a request. If this flag is set to "N" then OfficePath will generate a status and return it to the originator as soon as the request is written to the out-tray (or submitted to the internal-reader). If the application is, for example, going to update a database or write to an external mail network then you may want to design the application to only update the status when that process has completed. In this case, you should set the flag to "Y".

Note: Receipt acknowledgment status must always be returned by the application (if applicable) regardless of the setting of this flag.

Chapter 5. Creating an application to send mail

This is the simplest way to use OfficePath. If an application is designed only to send requests through OfficePath, without expecting any status information, no additional entries are needed on the OfficePath control file. The application can write its requests to one of the standard in-trays defined when OfficePath was installed.

An example of a request to send a document is shown in [Figure 3](#).

Figure 3. Request to send a document

```
Request header information

COMMAND   SEND      DOCUMENT
USER NAME OVERDUE   ACCOUNTS <=== Note 1
PASSWORD  SECRET
RECIPIENT P012345   PIPHOST   <=== Note 2
          P012346 PIPHOST
DOC NAME  List of overdue account at 01/11/90
DCA LEVEL 01403 <=== Note 3
REPLY     N <=== Note 4

Document contents

ACCOUNT   DESCRIPTION                               DATE DUE <=== Note 3
-----
M.A.S.H   500,000 Tongue depressors                 31/12/65
```

Note: The fields shown have the following functions:

1. This name will appear in the "Received from" field, in Personal Services. You may specify any name in this field.²
2. The name(s) of the person(s) to whom the mail is sent. You may specify up to 50 recipients. The recipients specified may include COPYLISTs which will be expanded by OfficePath.
3. The type of document which is being sent.
In this example, we use a "DCA LEVEL" of 1403 because the report, which is being sent, has been produced as a print file.
4. Status information is suppressed by setting the REPLY parameter to "N".

² If you are using the Batch interface, the user name and password will be validated by a password exit. The CICS interface does not perform any user validation; this is the responsibility of the application.

The request is sent into OfficePath using either the Batch or the CICS interface programs which will write it to one of the following datasets:

ZPMAILBX For Batch applications

ZPMAILSK For CICS applications

Application In-tray Your own application in-tray

Note: Use a different in-tray for each application you write, as this will ensure that applications are isolated from each-other, for example a "NOSPACE" condition on an in-tray will affect only one application. If you define your own application in-tray, then the OfficePath control file must be updated with the name of the in-tray.

Chapter 6. Obtaining status information from OfficePath

If it is important for your application to know that a document was successfully distributed to all its recipients, then you will need to request status information when you send the request to OfficePath. In this case, OfficePath will return confirmation messages to your application when the recipient has received his mail. Since the response is asynchronous, your application will need to maintain a log of the requests which it has sent and correlate the confirmations with the original request. To assist in this process, OfficePath allows the application to specify a correlation field, when it makes a request which is returned with the confirmation.

Two levels of status information are available. These are:

- DELIVERY** This status is returned by the OfficePath when the mail either arrives in the recipient's in-basket or when OfficePath determines, from its control file, that the mail cannot be delivered.
- RECEIPT ACK** This status is returned by the recipient's Office system when he first actions the item in his in-basket.

How to request that status information is returned

An example of a send which requests status information is shown in [Figure 4](#).

Figure 4. Sending a document and requesting status information

```
Request header information

COMMAND      SEND      DOCUMENT
USER NAME    OVERDUE   ACCOUNTS      <=== Note 1
PASSWORD     SECRET
RECIPIENT    P012345   PIPHOST
              P012346   PIPHOST
DOC NAME     List of overdue accounts at 01/11/90
DCA LEVEL
REPLY        Y          <=== Note 2
  TO SYS     REPDEST
  TO USER    OVERDUE   ACCOUNTS
RCPT ACK     N          <=== Note 3
CORRELATE    SEND0001  <=== Note 4
```


DIST CORR C0000000000001S

<=== Note 3,4

Document contents

ACCOUNT	DESCRIPTION	DATE DUE
M.A.S.H	500,000 Tongue depressors	31/12/65

Note: This format is similar to that shown in [Figure 3](#), but with additional parameters to handle the status information:

1. This name will appear in the "Received from" field, in Personal Services.³
2. Status information is requested by entering "Y" in the REPLY field. This requests that status information is returned to the application. The status information returned will indicate if the recipient was valid or not. When status information is requested, the following fields must also be supplied:

TO SYS This field specifies the destination record on the Control File which OfficePath is to use when writing the status information back for the application. This destination can be the same as the destination used to receive mail.

TO USER This field will be used as the "RECIPIENT" when the confirmation is returned to the outtray. It is a good idea to make this the same name as in the "USER NAME" parameter.

3. Receipt Acknowledgment is requested by entering a "Y" in the "RCPT ACK" field. If receipt acknowledgment is requested, then a status will be returned to the application the first time that the recipient actions the item in his in-basket. The "receipt acknowledgment" status will contain the value which was specified in the "DIST CORR" parameter.
4. Correlation information. These fields will be returned, un-modified, by OfficePath when a status document is returned to the application. They are intended to be used by the application to identify which request is being confirmed.

³ Receipt acknowledgment is returned as "special" mail items. This means that, in order for the application to receive it, the user must have a DEST record or type OUTTRAY or INTRDR which points to the application. If this is not case receipt acknowledgment is not generated and message PIPS053 is written to the CICS log.

The request is sent into OfficePath using either the Batch or the CICS interface programs which will write it to:

ZPMAILBX For Batch applications

ZPMAILSK For CICS applications

Application In-tray Your own application in-tray.

Note: Use a different in-tray for each application you write, as this will ensure that applications are isolated from each-other, for example a NOSPACE condition on an in-tray will affect only one application.

Format of the returned status information

An example of the status information returned is shown in [Figure 5](#).

Figure 5. Example of a status information document

```
Request header information
COD DOC SEND <=== Note 1
RECIPIENT OVERDUE ACCOUNTS <=== Note 2
DDN ACCOUNTS OVERDUE
CORRELATE SEND0001 <=== Note 3
NAK CODE 000000
DATE-TIME 01/11/1990 13:24:56 <=== Note 4
DEST-CODE XX

Document contents
RECIPIENT P012345 PIPHOST <=== Note 5
STATUS DELIVERED
RECIPIENT P012346 PIPHOST
STATUS **INVALID
```

Note:

1. The "COD DOC" field identifies this request as a status document.
2. The person to whom the status is returned.
This name is taken from the "TO USER" field in the original request.
3. Correlation information.
This value is taken from the "CORRELATE" field in the original request.

4. The date and time the status document was written to the out-tray.
5. Status information for the recipient(s).⁴

In this example we have shown how the status document would appear for the "Confirmation of Delivery" status. If your application requests "Receipt Acknowledgement" then a second status will be returned when the recipient first actions the document.⁵ This status differs slightly from the "Confirmation of Delivery" status as indicated below:

CORRELATE	This field would not be included
DDN	This field would not be included
DIST CORR	This field would be included, and would contain the value that was in the DIST CORR field of the original send request.
STATUS	The status will be "ACTIONED".

⁴ Since the status information is returned as soon as it is available, an application which sends to more than one recipient must allow for the status information being returned in one or more status documents.

⁵ The "Receipt Acknowledgment" status will normally be returned after the "Confirmation of Delivery" status; however since all status processing is asynchronous, your application should not assume that the statuses are returned in any particular order.

Chapter 7. Creating an application to receive mail

If an application is going to receive mail then you will have to define recipient(s), for the application, to whom mail can be addressed. All OfficePath recipients are identified using a two level addressing system - userid and address. Your application can be identified as either a single recipient (for example FRED at WITTON) or generically as a collection of recipients (for example any userid with address LONDON).⁶

Format of the output document

An example of the type of request which may be received is shown in [Figure 6](#).

Figure 6. Example of an output request

```
Request header information

DOCUMENT RECEIVED <=== Note 1
DOC NAME TEST DOCUMENT SENT TO AN APPLICATION
DCA LEVEL 01403 <=== Note 2
  CHARS
  LINES
REPLY Y <=== Note 3
  TO SYS DISOSSPS
  TO USER PSUSER PIPHOST
  RCPT ACK Y
SENT DATE 01/11/1990 10:50 <=== Note 4
RECIPIENT AUSER APPLN <=== Note 5
CORRELATE M.... <=== Note 6
DIST CORR C60B0F0B0D338BS..... <=== Note 6

Document contents

Sample text created in DW/370 <=== Note 2
```

⁶ Generally, you will use generic entries only if you are writing an interface to an existing mailing system. This reduces the maintenance required when a new user is defined to the mail system.

Note: In the example above, only the main fields are shown. Other fields, for example "SUBJECT", which may also appear are described in [Appendix B, "Parameters used for requests out of OfficePath"](#).

1. The "RECEIVED" field identifies the request as mail.
The field also determines what type of mail has been received. The values which may occur are:

DOCUMENT A document containing text in DCA defined format

PROFILE A reference to a document (no document text)

MESSAGE A message only

NOTE A note

2. A field which determines how to interpret the document contents.
If the document is in EDITABLE (DCA 1403) format then you will also receive the "CHARS" and "LINES" parameters which your application can use to determine the size of the document pages.⁷
3. Originator details.
The "TO USER" field contains the name (userid and address) of the person who sent the document. The "REPLY" and "RCPT ACK" fields show what status information was requested by the sender.⁸
4. The date and time that the request was made.
This indicates when the originator sent the document.
5. The person to whom the mail was addressed.
The application recipient to whom the mail as addressed.⁹
6. Correlation information.
These fields are used to return status information to the originator of a request. If your application is not designed to return status information, see ["Returns status information"](#), then you can ignore these.

⁷ OfficePath can perform document translations which ensure that any documents your application receives are in the format it is expecting.

⁸ Normally OfficePath will automatically generate the "delivered" status when the request is passed to your application; however you can override this and allow your application to return the status. This gives more flexibility to your application, but requires additional processing. See ["Returns status information"](#) for information about how to return status information.

⁹ Normally OfficePath will deliver the document to the application once for each recipient. It can, however, also be customized to deliver the document only once. In this case a list of recipients will be returned.



Chapter 8. Returning status information to OfficePath

An Application can specify that it will return status information to OfficePath for any mail which it receives. Normally OfficePath will generate a "delivered" status when it passes the mail to the application. However it can be customized to allow the application to generate a status. This means that an application can accept or reject mail items by setting a status in the originators status log. This is particularly useful for application which is an interface to other mail systems as it can return the 'delivered' or 'invalid' status once it has determined whether or not the user exists on that mail system. Note: OfficePath will never generate a "Receipt Acknowledgment" status. If your application does not generate this status, then any requests sent to it with RA will never be updated from "DELIVERED" to "ACTIONED".

An example of a status returned by an application is given in [Figure 7](#).

Figure 7. Example of how to return a status

```
Request header information

COMMAND  ACK      DELIVERED      <=== Note 1,4
USER NAME AUSER   APPLN          <=== Note 2
RECIPIENT PSUSER  PIPHOST       <=== Note 3
CORRELATE M....  <=== Note 4
DIST CORR C60B0F0B0D338BS..... <=== Note 4
```

Note: The following points should be noted:

1. This identifies the request as a status request.
The status returned may be one of the following values:

DELIVERED Recipient received mail
INVALID Recipient is not defined
CANCELLED Recipient cannot receive this type of mail
ERROR An error occurred processing this recipient
ACTIONED Recipient has worked with the mail item
2. This should contain the name (userid and address) of the user creating the status information.

This is the name of the recipient of the original mail item.

3. This should contain the name of the user to whom the status is being returned.
This is the name of the user who sent the original mail item.
4. The correlation field.
This field contains one of the correlation fields from the original mail item. The field which is used depends on the status being returned. Specifically, the "DIST CORR" field should only be used for the **ACTIONED** status, otherwise you should use the "CORRELATE" field.

Note: The **ACTIONED** status is returned only as the result of a "Receipt Acknowledgement" request.

PART 2 - BATCH INTERFACES

Chapter 9. Sending requests using the batch interface

This chapter describes how you can use the OfficePath Batch interface to enable your batch programs to send mail into the office system. The chapter describes how the interface operates, the definitions required to use it and explains the changes required in your batch procedures.

How the interface operates

The batch interface consists of a request formatting program and an input file, called the in tray. The format program takes a series of parameters, for example document name and recipient, and the document text from a sequential file and builds a request on the in tray file. The request is then read from the in tray by an OfficePath CICS transaction which periodically polls the in tray for new requests. Any requests found on the in tray are sent into the office system.

Each request on the in tray consists of the following types of record:

PROFILE This record contains all the parameters

TEXT These records contain the document text

TRAILER A record which indicates the end of the request

Input parameters are passed to the interface program in one of two ways:

Card Image The parameters are stored on a sequential dataset with one parameter per record. The parameter file is identified within the JCL by the DDname CARDFILE. The document, if any, is identified within the JCL by the DDname LISTDSN. Using this method only one distribution request can be made per program execution.

With document The parameters are stored with the document(s) to be processed. The document dataset is identified by the DDname LISTDSN. This method allows multiple requests to be processed by executing the program only once.

Defining a batch application

A Batch application can only be connected to OfficePath when the following functions have been performed:

1. An intray file has been created
2. An Intray record has been setup

You can have as many or as few intrays as you wish. For example, you may want to define an intray per application or have several applications share one intray.

Note: During the installation process a standard batch intray should have been created, known as the "Mailbox" (default DS name "PIPS.ZPMAILBX"). If you wish to use this file, then you can skip this section.

Creating an Intray file

The JCL shown in [Figure 8](#), and included in member PIPITRAY of dataset PIPS.JCL, can be used to define a new Intray file. The JCL will need customizing to your site standards.

Figure 8. The JCL to define an INTRAY file

```
//JOBNAMEF JOB (ACCOUNT), 'PROGRAMME',
//          MSGCLASS=A, CLASS=0, NOTIFY=XXXXXXX
//*****
//*
//* THIS JOB IS USED TO CREATE AND INITIALISE AN INTRAY FILE      *
//*
//*****
//STEP10   EXEC PGM=IDCAMS
//SYSPRINT DD  SYSOUT=*
//DD1      DD  UNIT=SYSDA, DISP=SHR,
//          VOL=SER=VOLSER      <== CHANGE TO YOUR INSTALLATION STANDARDS
//SYSIN    DD  *
    DELETE PIPS.VSAM.ITRAY FILE(DD1)
    DEFINE CLUSTER (NAME(PIPS.VSAM.ITRAY) -
                   VOL(VOLSER) -
                   FILE(DD1) -
                   FREESPACE(25 25) -
                   CYLINDERS(1 1) -
                   UNIQUE -
                   SHAREOPTIONS(2 3)) -
    DATA (NAME(PIPS.VSAM.ITRAY.DATA) -
          RECORDSIZE(2000 8192) -
          KEYS(12 0)) -
    INDEX (NAME(PIPS.VSAM.ITRAY.INDEX))
//*
```

```

//STEP20 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//INPUT DD DISP=SHR,DSN=PIPS.JCL(ZPMAIL1)
//OUTPUT DD DISP=SHR,DSN=PIPS.VSAM.ITRAY
//SYSIN DD *
REPRO INFILE(INPUT) OUTFILE(OUTPUT)
PRINT INFILE(OUTPUT)
//

```

The intray file now needs to be defined to CICS by an entry in the File Control table. A sample definition for this file is shown in [Figure 9](#) and included in member SAMPFCT2 on the PIPS.INSTALL library.

Figure 9. Sample Intray FCT definition

```

*****
*          FCT ENTRY FOR A BATCH INTRAY          *
*          (USED BY THE SAMPLE BATCH APPLICATION) *
*****
DEFINE FILE (ITRAYB)
  GROUP (OFFICEP)
  ADD (YES) BROWSE (YES) DELETE (YES) READ (YES) UPDATE (YES)
  RECORDFORMAT (V) STRINGS (1) RECOVERY (BACKOUTONLY)
  DSNAME (PIPS.VSAM.ITRAYB) STATUS (DISABLED) OPENTIME (FIRSTREF)

```

An entry must be put in the CICS JCL for this file if it is not allocated using the DSNAME option in the FCT. A sample definition is given below.

```

ITRAYB DD DISP=SHR,DSN=PIPS.VSAM.ITRAY

```

The INTRAY record

Each intray requires an intray record on the PIPCNTL Control File so that OfficePath will process the requests placed on the file by PIPGS3B.

The key of this record must match the dataset id defined in the FCT (for example ITRAYB).

Figure 10. The INTRAY maintenance panel

```
PIPB                O f f i c e P a t h  V e r s i o n  1.5.0          Date: 12/01/94
Function.... ENQUIRE          Control File Maintenance          Time: 09:25:45
Type..... INTRAY            Key.... ITRAYB
----- Input File Maintenance -----

Intray processed in batch..... Y

Comments.....
----- LAST MODIFIED BY ZMDVKD   AT TERM WL78ZMA7 AT 09:25:42 ON 12/01/94 -----
PIPCFM12 Enquiry only, input Function/Type or PF key
PF 1=HELP  4=EXIT  10=PREV REC  11=NEXT REC  12=QUIT
```

The only parameter that needs to be defined for this record identifies whether the intray is written by a CICS, or batch (or TSO) application. This must be set to "Y" for use with PIPGS3B. This will ensure that OfficePath, running in CICS, will issue an MVS enqueue before attempting to access the file. Since PIPGS3B also issues an MVS enqueue the integrity of the intray between batch and CICS is maintained. A value of "Y" also causes OfficePath, running in CICS, to close and disable the file after accessing the intray. This enables the batch application (PIPGS3B) to gain access whilst the CICS system is active.

Using the interface program

An example of the JCL for running PIPGS3B is shown in [Figure 11](#) and is included on PIPS.JCL as member PIPGS3B.

Figure 11. Example JCL for running PIPGS3B

```
//JOBNAMEX JOB (ACCOUNT), 'PROGNAME',
//          MSGCLASS=A, CLASS=0, COND=(0,NE), NOTIFY=XXXXXXXX
//*
//*          JOB TO TEST PRODUCT OPERATION
//*
//* YOU WILL NEED TO CHANGE THE VALUE OF THE 'RECIPIENT' PARAMETER
```

```

// * TO THE ID OF A USER WHICH IS VALID ON YOUR SYSTEM
// *
//SENDIT EXEC PGM=PIPGS3B, PARM='D'
//STEPLIB DD DISP=SHR, DSN=PIPS.LOAD
//ZPMAILBX DD DISP=SHR, DSN=PIPS.VSAM.ZPMAILBX
//PIPCNTL DD DISP=SHR, DSN=PIPS.VSAM.PIPCNTL
//SYSABEND DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//LISTDSN DD DISP=SHR, DSN=PIPS.JCL (TESTNOTE)
//CARDFILE DD *
COMMAND SEND NOTE
DOC NAME NOTE 07/09/90 12:24:00 TSO
SUBJECT INSTALLATION VERIFICATION DOCUMENT
AUTHOR C M LANGFORD
DOC DATE 07/09/90
REPLY N
RECIPIENT XXXX PIPHOST
// *
//ENDIT EXEC PGM=IEFBR14
//

```

The JCL shown above will need tailoring to meet your installation standards.

The supplied JCL shows how to send a single document using the interface program. The parameters are taken from the CARDFILE dataset and the document contents from the LISTDSN dataset. You could, for example, modify your overnight report jobs to output to a temporary dataset (rather than SYSOUT) which would then be used as the LISTDSN dataset. This gives you a simple mechanism for distributing reports to manager's inbaskets.

Note: If you want to send a message or an acknowledgment only, then you should remove the PARM='D' option. In this case the program will not access the LISTDSN dataset.

[Input Parameters](#)

The parameters are passed to PIPGS3B either on a cardfile, with one parameter per record, or with the document to be processed. The parameters which may be entered and the validation performed is described in detail in [Appendix A, "Parameters used for requests into OfficePath"](#).

[Document dataset](#)

The document dataset may be any sequential file or member of a PDS. OfficePath will examine the DCB information to determine how to process the input dataset. If the DCB indicates that the dataset contains either ANSI-defined or Machine print controls, then it will respond to these when processing the document. This means that, for example, batch reports can be sent through OfficePath and the pagination will be preserved.

Return codes generated

The return codes produced by program PIPGS3B are described in the *OfficePath Reference and Diagnosis* manual.

Sending multiple documents

The JCL required to send multiple documents into OfficePath is similar to that shown in [Figure 11](#), except that the PARM field must have the value "P". This indicates that the parameter data for each document is included in the dataset identified by DDname LISTDSN. A set of records should be built up on this file, for each document, which must be as follows:

1. Low-values record (35 bytes minimum)
2. Parameter records (one parameter per record)
3. High-values record (35 bytes minimum)
4. Document records (optional)

Note: If the file is defined with ANSI-defined or Machine controls, then any parameters supplied must also include a printer-control character.

If you are sending a message or acknowledgment only, then you should only supply the low-values and parameter records.

Further considerations

The following points need to be noted:

1. ANSI files - these will have blank lines inserted by OfficePath in response to the ANSI-defined print controls. This can result in a page being too large for a single in-tray record and OfficePath having to split it across records.
 2. Variable length files - the interface processes these as if they were fixed-length with a record-length equal to the largest found in the first 1500 records on the file. This value is used for all documents on the file, so a particular document may have lines longer than expected.
 3. If the interface program is called from another program the calling program should not specify the "de-allocate at close" ("DALCLOSE" in DYNALLOC) option for the in-tray file.
-

Example application

Member SAMPGS3B in the PIPS.JCL library contains an example of how PIPGS3B can be used with an existing application. In the JCL the IDCAMS program is invoked with a "listcat" command. The output from the IDCAMS program is written to a temporary file which is used as the input document for the PIPGS3B program. The program reads the instream parameters and sends the document to the recipient's inbasket. Using this mechanism any report program could easily be amended to route the output to an inbasket, rather than printing it and distributing it using the internal mail.

Chapter 10. Automating the distribution of Pre-printed Forms

A batch program is supplied that will take as input a print-stream containing pre-printed forms (for example statements, invoices, or letters) and convert them into a format that can be sent to Email users via OfficePath.

The program requires input in the form of a sequential data set containing a 1403 print-stream with ANSI-defined machine control characters (RECFM=FBA, LRECL=133).

AUTOSEND assumes that each form in the print-stream is made up of a front page containing the information required to distribute the item, this may then be followed by further pages. The distribution information, such as the recipient-id, must always appear in the same position on the first page of every item. A control data set must be created containing the position and length of each of the required fields. If any errors are found with the control parameters then the program will terminate with a suitable error message and the documents will not be processed.

AUTOSEND produces a sequential data set which contains the documents in a format that is ready for distribution via the OfficePath batch interface program PIPGS3B.

If the Nickname parameter is used instead of the Recipient parameter then a sub program (AUTONICK) is called which reads a file to "convert" each recipient nickname into their user-id and node.

Creating the files

The AUTOSEND program requires a sequential file, which is used to report distribution errors, and a VSAM file, which is used for nickname validation.

In addition a file, AUTOCNTL, of Control Parameters is required. In the examples which follow this is shown as in-stream data.

AUTOERR file

This is a sequential file which is used to report distribution errors. This file can be created using the standard ISPF dialogs (The Utility Selection Menu - option 2). The file is accessed from a batch job which is submitted by OfficePath. The DCB information for the file must, therefore, match the DCB for the OfficePath internal reader.

Any name can be used for this file. However, the name chosen must match the name specified on the NAK-DSN parameter (see ["Using the AUTOSEND JCL"](#)).

This file is optional. If the file is not created, then no errors can be reported.

PIPANICK file

This is a VSAM KSDS file which is used to validate nicknames. Sample JCL to create the file can be found in **PIPANICK** in the **PIPS.JCL** library. The JCL is shown in [Figure 12](#).

Member PIPANIC1 should be edited to contain a list of your nicknames with their corresponding user-id's before running PIPANICK.

This file is only required if the **NICKNAME** option is used (see ["Using the AUTOSEND JCL"](#)).

Figure 12. JCL to create PIPANICK file

```
//JOBNAME1 JOB (ACCOUNT), 'PROGRAM',
//          CLASS=0,MSGCLASS=A,NOTIFY=XXXXXXXX
//*****
//*
//* THIS JOB IS USED TO CREATE AND INITIALISE THE PIPANICK FILE      *
//*
//* THE PIPANIC1 MEMBER WILL NEED TO BE EDITED BEFORE RUNNING THIS *
//* JOB.  THE PIPANIC1 MEMBER CONTAINS THE LIST OF NICKNAMES WHICH *
//* THE AUTOSEND PROGRAM CAN VALIDATE AND THEIR ASSOCIATED USERIDS *
//* AND ADDRESSES.                                                 *
//*
//* NOTE: THIS JOB USES TWO REPRO STATEMENTS. THIS FIRST REPRO LOADS *
//*       THE FILE, THE SECOND COPIES THE RECORDS OVER.           *
//*****
//STEP10   EXEC PGM=IDCAMS
//SYSPRINT DD  SYSOUT=*
//LOADFILE DD  DISP=SHR,DSN=PIPS.JCL(PIPANIC1)
//DD1     DD  UNIT=SYSDA,DISP=SHR,
//        VOL=SER=VOLSER      <== CHANGE TO YOUR INSTALLATION STANDARDS
//SYSIN   DD  *
           DELETE PIPS.VSAM.PIPANICK FILE(DD1)
           DEFINE CLUSTER (NAME(PIPS.VSAM.PIPANICK) -
                           VOL(VOLSER) -
                           FILE(DD1) -
                           TRACKS(1 1) -
                           UNIQUE -
                           SHAREOPTIONS(2 3)) -
           DATA (NAME(PIPS.VSAM.PIPANICK.DATA) -
                 RECORDSIZE(80 80) -
                 KEYS(56 0)) -
           INDEX (NAME(PIPS.VSAM.PIPANICK.INDEX))

           REPRO INFILE(LOADFILE) ODS('PIPS.VSAM.PIPANICK') SKIP(16) COUNT(1)

           REPRO INFILE(LOADFILE) ODS('PIPS.VSAM.PIPANICK') SKIP(17)
//
```

Using the AUTOSEND JCL

An example of a job steps required to run the AUTOSEND program can be found in the **PIPS.JCL** library, member **SAMPASND**. The job contains two steps:

AUTOSEND Executes the AUTOSEND program which formats the documents.

SEND Executes the OfficePath batch interface program to distribute the documents.

An example of a job to setup the nickname file can be found in the **PIPS.JCL** library, member **NICKNAME**.

Figure 13. Job steps required to execute the AUTOSEND program

```
/******  
/*  
/*          AUTOSEND FUNCTION          *  
/*  
/* THIS PROGRAM IS USED TO 'SPLIT' A STANDARD BATCH REPORT INTO A *  
/* NUMBER OF DISTRIBUTION REQUESTS. THE RECIPIENT NAMES ARE TAKEN *  
/* FROM DATA IN THE REPORT. THESE JOBSTEPS ARE INTENDED TO BE *  
/* APPENDED TO THE END OF EXISTING BATCH REPORT JCL. THE ACTUAL *  
/* PROGRAM WHICH PRODUCES THE REPORT DOES NOT REQUIRE ANY CHANGES. *  
/*  
/* THE BATCH REPORT JCL REQUIRES THE FOLLOWING CHANGES: *  
/* 1 - CHANGE THE JCL SO THAT THE REPORT IS ROUTED TO A TEMPORARY *  
/* DATASET RATHER THAN TO A SYSOUT FILE. FOR EXAMPLE, *  
/* CHANGE *  
/* OREPORT DD SYSOUT=D *  
/* TO *  
/* OREPORT DD DISP=(NEW,PASS),DSN=&REPORT, *  
/* DCB=(RECFM=FBA,LRECL=133,BLKSIZE=BLKSIZE), *  
/* SPACE=(CYL,(2,1)),VOL=SER=SYSDA *  
/* 2 - APPEND THESE STEPS TO THE END OF THE JCL *  
/* 3 - CUSTOMISE THE AUTOCNTL PARAMETERS TO MATCH YOUR REQUIREMENTS *  
/******  
/*  
/* STEP 1 - CONVERT THE PRINT FILE INTO A SERIES OF REQUESTS *  
/*  
/*AUTOSEND EXEC PGM=AUTOSEND,TIME=(,10)  
/*STEPLIB DD DISP=SHR,DSN=PIPS.LOAD  
/*AUTONICK DD DISP=SHR,DSN=PIPS.VSAM.PIPANICK  
/*AUTOOUT DD DISP=(NEW,PASS),DSN=&LISTDSN,  
/* DCB=(LRECL=133,RECFM=FBA,BLKSIZE=6118),  
/* SPACE=(6118,(10,1)),UNIT=SYSDA  
/*SYSOUT DD SYSOUT=*  
/*AUTOIN DD DISP=SHR,DSN= <=== PUT YOUR DATASET HERE  
/*AUTOCNTL DD *  
* FIXED PARAMETERS  
CHARS PER LINE  
*DEF-NODE DEFNODE1  
HEADER N  
INDENT
```

```

MESSAGE      PLEASE REVIEW THIS DOCUMENT AND
MESSAGE      RETURN TO ME WITH ANY QUESTIONS.
NAK-DSN      PIPS.AUTOERR
*PASSWORD    RSW
USER         RSW
* POSITIONAL  PARAMETERS
DATE         2,64,8
*DOC-NAME    2,28,20
NICKNAME     1,11,20
*NODE        1,11,8
PAGE         4,64,4
*RECIPIENT  1,11,8
SUBJECT      2,28,10
//*
//* STEP 2 - INVOKE PROGRAM TO SEND REQUESTS
//*
//SEND       EXEC PGM=PIPGS3B,PARM='P',COND=(7,LT,AUTOSEND)
//STEPLIB   DD DISP=SHR,DSN=PIPS.LOAD
//ZPMAILBX  DD DISP=SHR,DSN=PIPS.VSAM.ZPMAILBX
//PIPCNTL   DD DISP=SHR,DSN=PIPS.VSAM.PIPCNTL
//LISTDSN   DD DISP=(OLD,DELETE),DSN=&LISTDSN
//CARDFILE  DD DUMMY
//SYSOUT    DD SYSOUT=*
//

```

[AUTOCNTL Control Parameters](#)

Note: Any parameter found that contains an asterisk or a space as the first character will be ignored. This may be used for comments or temporary changes.

The following parameters can be defined in the control data set to control the processing of OfficePath:

Name	Format	Required	Description
CHARS PER LINE	numeric	N	Defines the number of characters per line of the input print-stream. If supplied it must be a value between 1 and 132. If not supplied the default will be 80 plus the value of the indent parameter.
DEF-NODE	text (8)	N	Defines a default node name that is used in conjunction with the recipient parameter. This may be useful if used with the OfficePath copylist function to allow distribution to a list of people. To use this facility set the default node to "COPYLIST".
HEADER	Y/N/T	N	Defines whether the first page of each item, containing the distribution parameters, will be sent

			along with the document or if AUTOSEND is being run in test mode. If supplied it must be "Y", "N" or "T". If not supplied the default will be "Y". Header must be set to "Y" if no page parameter is defined. If header is set to "T" then the program examines the first page, extracts the fields defined by the other control parameters, displays the values it has found and then terminates without processing any items. Test mode is useful when first setting up the AUTOSEND function to check that fields are being "picked up" correctly.
INDENT	numeric	N	Defines the number of characters that each line will be shifted right in order to centralize the report. If supplied it must be a value between 1 and 132. If not supplied then the default will be
MESSAGE	text (64)	N	Up to four lines of message text that will accompany the document.
NAK-DSN	dataset	Y	Defines the name of a sequential data set that will be used to store any negative acknowledgements received during the distribution of the documents.
USER	text (8)	N	Defines the originating user id. If not supplied then it will default to "AUTOSEND".
PASSWORD	text (8)	N	The password for the userid specified in USER.

Note:

- The as-shipped version of OfficePath will override any supplied userid with the userid which is assigned to the job. This can be amended, if required, by changing the **PIPUSRCK** program. Instructions for changing this program can be found in the *OfficePath Installation and Administration* guide.
- The following parameters are used to determine the position of fields in the front page of each distribution. All the parameters use the format **rrr,ccc,lll**:
 - rrr defines the row (line) containing the field (machine control characters should be taken into account when calculating the row number).
 - ccc defines the starting column of the field.
 - lll defines the length of the field.

Name	Format	Required	Description
DATE	rrr,ccc,lll	N	The position and length of the date.
DOC-NAME	rrr,ccc,lll	N	The position and length of the Document Name. If this

			parameter is not supplied then it will default to "AUTO/recipient/date/nnnnn" if the recipient parameter is used or to "AUTO/nnnnn/date" if the nickname parameter is used (where nnnnn is a one up sequence number incremented for each distribution).
NICKNAME	rrr,ccc,lll	N	The position and length of a Nickname used to determine a recipient and node. Mandatory if no Recipient parameter supplied.
NODE	rrr,ccc,lll	N	The position and length of the Recipients Node.
PAGE	rrr,ccc,lll	N	The position and length of the Page Number. The column number must be the left most position of largest possible page number. Each document is assumed to commence on page 1. If not supplied then each document is assumed to be one page only.
RECIPIENT	rrr,ccc,lll	N	The position and length of the Recipients ID. If the recipient and nickname parameters are both used then the recipient field is checked first.
SUBJECT	rrr,ccc,lll	N	The position and length of the Documents Subject.

Note: A CORRELATE parameter is automatically created for each distribution request in the following format: "AUT/nnnnn/date" where "nnnnn" is a one up sequence number that corresponds to the number of the item in the print-stream. If a distribution fails a negative acknowledgement (NAK) will be written to the dataset indicated by the NAK-DSN parameter. Each NAK will contain the correlation field so that the original distribution can be checked.

[AUTOSEND Messages](#)

The following message may be produced by OfficePath:

AUTO001 ERROR PERFORMING A xxxxxxxx ON FILE xxxxxxxx - STATUS NN

- An error occurred accessing the indicated file. The file status causing the error is displayed at the end of the message. This may be caused by a missing file or an incorrect definition in the JCL.

AUTO002 INVALID FORMAT FOUND FOR PARAMETER xxxxxxxx

- A parameter has been found that is in an unexpected format or has one or more values missing.

AUTO003 INVALID LENGTH FOUND FOR PARAMETER xxxxxxxx

- A field position parameter has been found that has a missing or invalid length value.

AUTO004 INVALID COLUMN POSITION FOUND FOR PARAMETER xxxxxxxx

- A field position parameter has been found that has a missing or invalid column value. This may be due to the column position being larger than the number of characters per line.

AUTO005 INVALID ROW POSITION FOUND FOR PARAMETER xxxxxxxx

- A field position parameter has been found that has a missing or invalid row value.

AUTO006 COLUMN PLUS LENGTH MUST BE LESS THAN 133 FOR
PARAMETER xxxxxxxx

- A field position parameter has been found that has a column position and a length that add up to more than the maximum number of characters per line.

AUTO007 UNRECOGNISABLE PARAMETER FOUND - xxxxxxxx

- A parameter has been found that does not match any known parameters.

AUTO008 PARAMETER RECIPIENT OR NICKNAME MUST BE SUPPLIED

- A position parameter must be supplied for the Recipient or for Nickname. These fields define where the document is sent to, so are mandatory.

AUTO009 PARAMETER NAK-DSN MUST BE SUPPLIED

- NAK-DSN is a mandatory parameter that defines the sequential data set that is used to store any distribution errors.

AUTO010 HEADER PARAMETER MUST BE Y, N OR T

- The Header parameter has been supplied, but has been defined as something other than Y (Yes - include header page), N (No - do not include header page) or T (Test - test field positions in the header).

AUTO011 HEADER PARAMETER MUST BE Y IF USING SINGLE PAGE FORMS

- The Header parameter has been supplied set to No and no page parameter has been supplied. If no page parameter has been supplied this indicates that there is only one page per distribution (that is the header). Set the Header parameter to Y or include a page parameter.

AUTO012 xxxxxxxxxxxx SENT TO yyyyyyy-zzzzzz

- This is an information message confirming the distribution request. Where "xxxxxx" is the document name, "yyyyyy" is the recipient and "zzzzzz" is the recipients node.

AUTO994 REPORT CREATED CONTAINING NO INFORMATION AND HEADER PAGE OMITTED

- An item in the print stream has been found that has a header page only and the header parameter has been set to "N". The item is distributed containing the above message.

AUTO995 ITEM nnnnn UNABLE TO FIND NICKNAME xxxxxxxxxxxx

- An item in the print stream has been found that has a nickname that does not exist on the nickname file. The item is not distributed and processing continued with the next item.

AUTO996 ITEM nnnnn NOT IN RECOGNISED FORMAT

- An item in the print stream has been found that does not have the mandatory fields defined in the header or they are in a different position to those defined by the position parameters. The item number will determine which item in the stream failed. If the indicated item is the first in the stream and no other items have been processed then this would indicate a problem finding the page numbers.

AUTO997 ERROR DETECTED - AUTOSEND TERMINATING

- An error was detected with one or more of the control parameters so OfficePath terminates and does not process any distributions. The parameters in error will be listed before this message.

AUTO998 PROCESSING COMPLETED - RETURN CODE nn

- AUTOSEND PROCESSING HAS FINISHED, THE RETURN CODE IS AS FOLLOWS:
 - 0** All requests have been processed
 - 4** Some requests have been processed - at least one was rejected
 - 8** No requests were successful
 - 12** A serious error prevented processing

AUTO999 PROCESSING COMPLETED - nnnnn ITEMS PROCESSED, nnnnn ITEMS REJECTED

- OfficePath has processed all items in the print-stream. This message contains a total number of items processed and the number of items rejected.
-

Chapter 11. Outbound - using a VSAM file

The OfficePath Batch interface can also be used to receive mail from office systems users. This allows you to construct applications which are driven by the user simply sending a document or message to a "special" userid. Alternatively, you could construct an interface to an existing mail system using the send and receive capabilities of the Batch interface. This chapter describes what definitions are necessary to allow a batch application to receive mail. It also describes how the mail interface works and how to call the interface program.

How the interface operates

The interface consists of a file, called the outtray, to which OfficePath writes requests sent to the batch application and a batch program to interpret the file. When a document is sent to the "special" userid, OfficePath writes the document to the outtray file, and optionally submits JCL to invoke the application. The JCL is submitted after a time interval has expired rather than immediately, any other requests which arrive in this time will not generate a new JCL request. This allows you to "batch-up" requests so that only one job is submitted rather than one job per request.

When the application executes it invokes an OfficePath program which interprets the request on the outtray and converts it into a series of parameters and document text on a sequential file. The application can then read the sequential file to process the document. The interface program can be invoked using JCL or from within a TSO CLIST.

Note: OfficePath can also generate batch requests using the JES internal reader with instream data (see "[Outbound - using an Internal Reader](#)"). The internal reader method is more flexible and allows parameters to be imbedded in the JCL to modify its action. However, the outtray method allows a single batch job to process many mail items.

The document profile information may be output as parameters in one of two ways:

Card Image The parameters are written to a sequential dataset with one parameter per record. The parameter file is identified by the DDname CARDFILE.

With document The parameters are written with the document to a dataset identified by the DDname LISTDSN. The parameter records precede the document records for the item.

Defining an Outbound destination

A batch application can only receive distributions from OfficePath when the following functions have been performed:

1. An outtray has been created.
2. A Destination record, of type OUTTRAY, has been added to the control file.
3. A User record has been added to the control file.
4. A JCL record has been added to the control file (This JCL record is optional).

Note: Example definitions to carry out steps 2 - 4 are contained in member PIPCTL4 of the PIPS.JCL library. These may have been setup when the PIPCTL job was run at installation time.

Creating an outtray file

The JCL shown in [Figure 14](#) and included in PIPS.JCL member PIPOTRAY can be used to define a new outtray file.

Figure 14. The JCL to define an Outtray file

```
//JOBNAMEX JOB (ACCOUNT), 'PROGRAMME ',
//          MSGCLASS=A, CLASS=0, NOTIFY=XXXXXXXX
//*****
//*THIS CREATES AND INITIALISES AN OUTTRAY FILE                               *
//*****
//STEP10   EXEC PGM=IDCAMS
//SYSPRINT DD  SYSOUT=*
//DD1      DD  UNIT=SYSDA, DISP=SHR,
//          VOL=SER=VOLSER
//SYSIN    DD  *
           DELETE PIPS.VSAM.OTRAY FILE(DD1)
           DEFINE CLUSTER (NAME(PIPS.VSAM.OTRAY)           -
                           VOL(VOLSER)                   -
                           FILE(DD1)                     -
                           CYL(1 1)                      -
                           SHAREOPTIONS(2 3)              -
                           UNIQUE)                       -
           DATA (NAME(PIPS.VSAM.OTRAY.DATA)             -
                 RECORDSIZE(8192 8192)                  -
                 KEYS(12 0))                             -
           INDEX (NAME(PIPS.VSAM.OTRAY.INDEX))
/*
//STEP20   EXEC PGM=IDCAMS
//SYSPRINT DD  SYSOUT=*
//INPUT    DD  DISP=SHR, DSN=PIPS.JCL(ZPMAIL1)
//OUTPUT   DD  DISP=SHR, DSN=PIPS.VSAM.OTRAY
//SYSIN    DD  *
           REPRO INFILE(INPUT) OUTFILE(OUTPUT)
           PRINT INFILE(OUTPUT)
//
```

This file now needs to be defined to CICS by an entry in the File Control table (FCT). A sample definition is presented in [Figure 15](#), and is included in member SAMPFCT4 in the PIPS.INSTALL library.

Figure 15. Sample FCT entry for an Outtray file

```
*****
*          FCT ENTRY FOR A BATCH OUTTRAY          *
*          (USED BY THE SAMPLE BATCH APPLICATION)  *
*****
DEFINE FILE (OTRAYB)
  GROUP (OFFICEP)
  ADD (YES) BROWSE (YES) DELETE (YES) READ (YES) UPDATE (YES)
  RECORDFORMAT (V) STRINGS (1) RECOVERY (BACKOUTONLY)
  DSNAME (PIPS.VSAM.OTRAYB) STATUS (DISABLED) OPENTIME (FIRSTREF)
```

An entry for this file must be put in the CICS JCL if it is not allocated using the DSNAME option in the FCT. A sample entry is shown below:

```
OTRAYB DD DISP=SHR,DSN=PIPS.VSAM.OTRAY
```

Destination record, OUTTRAY type

An example of the panel used to add a new outtray definition is shown in [Figure 16](#). To reach this panel, enter the online maintenance suite with transaction "PIPX" or the customized equivalent. From this panel, type "INSERT" as the function, "DEST" as the type and enter a key for the record that you are adding. This key can be up to 8 characters in length, and is used only for internal processing.

The online suite allows the first character of any command to be entered in the function and type fields. The relevant program will then pad the field to the full name.

Once the program has responded, overtype the "DESTINATION TYPE" field with "OUTTRAY" and press enter. This will display all of the fields that are valid for this type of record.

Once one record of the correct type is entered on the control file, it is possible to clone a new record using the existing record. This is done by displaying the existing record on the panel (using a function of "ENQUIRE" and key of the existing record), and then overtyping the function to "INSERT" and key to the new key. This creates a new record which can then be amended as required.

Figure 16. The Destination panel for OUTTRAY type destinations

```

PIPG                O f f i c e P a t h  Version 1.5.0          Date: 12/01/94
Function.... ENQUIRE          Control File Maintenance      Time: 11:43:49
Type..... DEST                Key.... FILEB
----- Destination Maintenance -----
          Destination type: OUTTRAY
DOCUMENT TYPES                                APPLICATION ATTRIBUTES
Accept RF Text..... N                    Accumulate recipients..... N
Accept FF Text..... N                    Can process ASA controls..... N
Accept PC File..... N                    Can process NOTE doctype..... Y
Accept Image..... N                      Uses interface program..... Y
Accept Editable..... Y                   Returns status information... N
Accept Msg/External.. Y                 Outtray processed in batch... Y
Other document types.                    Line-length.....
.....

CHARACTER SET                                CICS PARAMETERS
Accept GCID..... MULTI-LINGUAL          Output file name..... OTRAYB
                                          Mail transaction.....
                                          and terminal (optional).....

Comments.....
----- LAST MODIFIED BY ?????????? AT TERM ?????????? AT ?????????? ON ?????????? -----
PIPCFM12 Enquiry only, input Function/Type or PF key
PF 1=HELP  4=EXIT 10=PREV REC 11=NEXT REC 12=QUIT

```

Description of fields

Default values, where applicable, are indicated by brackets.

- Document Types

This set of parameters identifies the types of documents that this destination can receive. The value of the parameter is either "Y" to receive this type of document or "N" to disqualify it. If a document is sent that cannot be received by this destination, then OfficePath will attempt to convert the document to a type which the destination can accept, using a formatter program. If no suitable formatter program can be found, then the request is deleted and a **CANCELLED** status is returned to the originator of the distribution. Details of the supplied formatter programs can be found in the *OfficePath Installation and Administration* manual.

- Application Attributes

This set of attributes determines how OfficePath will format any mail that is sent to this destination. The recommended values, which may be given below, are for applications which you write. If you are installing an Keane Ltd product (for example, FAX), you should refer to its installation instructions for details of the settings which are required.

- Accumulate recipients
If a document is to be sent to multiple recipients on the same destination OfficePath can either deliver the request as one document with a list of recipients, or as one document per recipient. If this flag is "Y" then OfficePath will deliver the request as one document with a list of recipients. The RECIPIENT field of the profile will list all of the recipients, on this destination, to whom the document was distributed. If this flag is not set, then OfficePath will deliver multiple documents, each with one recipient.
- Can process ASA controls
EDITABLE documents contain an ANSI-defined Print Control character (ASA) at the start of each line (for example "-", which means "advance two lines before printing"). Some user applications do not respond to these controls, resulting in bold lines (using the overstrike control "+") being processed twice. If the ASA flag is "N" then OfficePath will interpret the ANSI-defined controls (remove overstruck lines and insert blank lines) before passing the document to the application.
- Can process NOTE doctype
If an application cannot receive notes, OfficePath can alter the control information associated with a note to make it look like an EDITABLE document. With this processing, notes can be passed to applications that cannot normally cope with this type of distribution.
- Uses interface program
If the outtray is to be processed by either the CICS or Batch interface programs then OfficePath needs to be made aware of this. You should always set this to "Y" for any applications you write.

Note: If this flag is set to "N", OfficePath will write to the outtray using another (older) format which will not be recognized by the interface program.

- Returns status information
If the application creates status messages (ACK requests) to indicate to the originator the outcome of the processing, then this flag should be set to "Y". OfficePath will return the status generated by the application to the originator. If this flag is set to "N", then OfficePath does not wait for the application to return a status but instead returns a status of **DELIVERED** when the request has been copied onto the outtray.

Note: The **ACTIONED** (Receipt acknowledge) status must always be generated by the application, regardless of the setting of this flag.

- Outtray processed in batch
This parameter needs to have a value of "Y" for this application. This will ensure that OfficePath issues an MVS enqueue before attempting to access the file. This is to ensure that the integrity of the Outtray is maintained between the CICS and Batch programs. The value of "Y" also causes OfficePath, running in CICS, to close and disable the file after accessing the outtray.

- Line length
The line length of the data that is written to this file can be altered to match the line length that the application/user is expecting. If the document text on a line exceeds this value, it will be truncated.

Note: Documents which have been processed by the formatter programs will have a maximum of 132 characters of text per line (plus 1 byte for the ANSI-defined control character).

- Character Set

OfficePath can present distributed text to an application in a GCID that the application is expecting. OfficePath will convert the contents of the profile and print-image text to the character set and code page specified by the GCID value. Valid values for this are:

- "LOCAL" - Use the value from the SYSTEM record

This value should have been set to the GCID of your country during the installation process.

- "MULTI-LINGUAL" - The multi-lingual GCID defined by IBM

This is the GCID used internally by Personal Services and DISOSS.

- One of the values in [Appendix C, "Document type and Character Set Code Numbers"](#).

Note: If the document is of type RFT or FFT it will remain in multi-lingual in order to be processed by the DisplayWrite family of products.

- CICS Parameters

- Output file name

This identifies the CICS id of the outtray. The id specified here must match the dataset name defined in the FCT.

- Mail transaction

This field should be left blank.

- Termid

This field should be left blank.

User record

You must define one or more user records for every destination record. The user records can be defined explicitly or generically.

The User record panel is obtained from the main menu by typing a function of "INSERT", a type of "USER" and a unique key that will match the recipient to whom you will address the distribution. You can generically define a group of users such as OUTTRAYB-* so that mail sent to any users on node "OUTTRAYB" will have mail routed to the destination to which this record points at. An example panel is shown in [Figure 17](#).

In order to define users explicitly, you must add user records keyed on both node and userid, separated by a hyphen. For example, OUTTRAYB-USER1 could point at FILEB1 and OUTTRAYB-USER2 could point at FILEB2.

The destination that is specified in this record must match a destination record that already exists on the control file.

Figure 17. The User maintenance panel

```
PIPA                O f f i c e P a t h  V e r s i o n  1.5.0          Date: 12/01/94
Function.... ENQUIRE          Control File Maintenance          Time: 09:24:08
Type..... USER              Key.... OUTTRAYB-*
----- User Maintenance -----

Destination..... FILEB

Comments.....
----- LAST MODIFIED BY ZMDVKD   AT TERM WL78ZMA7 AT 09:24:06 ON 12/01/94 -----
PIPCFM12 Enquiry only, input Function/Type or PF key
PF 1=HELP  4=EXIT  10=PREV REC  11=NEXT REC  12=QUIT
```


JCL record

OfficePath can submit JCL to process any information that appears on a batch outtray, if that JCL is found on the control file. If there is no JCL to process the outtray, the information will remain on the file. The JCL records are set up using the on-line maintenance suite, by typing a function of "INSERT", a type of "JCL" and a key that corresponds to the destination record set up for this application.

The panel shown in [Figure 18](#) is the JCL record that is supplied to process requests sent to a batch destination FILEB. This record may have been set up when the PIPCTL job was run at installation time.

Figure 18. The JCL maintenance panel

```
PIPC                               O f f i c e P a t h                               Date: 19/09/95
Function.... ENQUIRE               Control File Maintenance                       Time: 08:51:47
Type..... USER                     Key.... FILEB
----- JCL Maintenance -----
***** TOP OF DATA *****
//*
//* STEP10 WILL RUN THE PIPGT3B PROGRAM
//*
//STEP10   EXEC PGM=PIPGT3B, PARM='DM'
//STEPLIB DD DSN=PIPS.LOAD, DISP=SHR
//OUTTRAY DD DSN=PIPS.VSAM.OTRAYB, DISP=SHR
//SYSOUT  DD SYSOUT=*
//SYSABEND DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//LISTDSN DD SYSOUT=*, DCB=(RECFM=FB, LRECL=133)
//
***** BOTTOM OF DATA *****

----- LAST MODIFIED BY                AT TERM WU78PDA4 AT 12:22:49 ON 01/12/93 -----
PIPCFM12 Enquiry only, input Function/Type or PF key
PF 1=HELP  4=EXIT  7=BACKWARD  8=FORWARD  10=PREV REC  11=NEXT REC  12=QUIT
```

This panel allows extra lines of JCL to be inserted using the (Insert) line command "In", where n is an integer indicating the number of lines to be inserted. The additional lines will be placed after the current line. Individual lines can be deleted using the "D" (Delete) line command or repeated using the "R" (Repeat) line command. Each JCL record can contain a maximum of 56 lines of JCL. This record is described in more detail in ["The JCL records"](#).

SYSTEM record

This record contains a field called the "Batch JCL submit interval" which is used to control the time delay in submitting the Batch JCL. This record is described in the *OfficePath Installation and Administration Manual*.

Using the interface

PIPGT3B can be executed as a batch job or invoked within a TSO CLIST.

An example of the JCL for running PIPGT3B is shown in [Figure 19](#) and is included in member SAMPGT3B on the PIPS.JCL library.

Figure 19. Example JCL for running PIPGT3B

```
//JOBNAMEX JOB (ACCOUNT), 'PROGRAMME ',  
//          MSGCLASS=A, CLASS=0, COND=(0,NE), NOTIFY=XXXXXXXX  
//*  
//*          JOB TO READ A DOCUMENT FROM A BATCH OUTTRAY  
//*  
//RECEIVE EXEC PGM=PIPGT3B, PARM='S'  
//STEPLIB DD DISP=SHR, DSN=PIPS.LOAD  
//OUTTRAY DD DISP=SHR, DSN=PIPS.VSAM.PIPOTRAY  
//SYSABEND DD SYSOUT=*  
//SYSUDUMP DD SYSOUT=*  
//SYSOUT DD SYSOUT=*  
//CARDFILE DD SYSOUT=*, DCB=(RECFM=FB, LRECL=133)  
//LISTDSN DD SYSOUT=*, DCB=(RECFM=FB, LRECL=133)  
//*  
//ENDIT EXEC PGM=IEFBR14  
//
```

The supplied JCL shows an example of the PIPGT3 program reading a single request from the outtray. The parameters are written to the CARDFILE dataset and the document contents to the LISTDSN dataset. The request is deleted after processing. The processing performed by the program can be modified by altering the PARM statement as described in ["Execution Parameters"](#).

Execution Parameters

The default action (if no parameter is specified) is for the program to process and delete all requests on the outtray. The parameters are written to the CARDFILE dataset with individual requests separated by a low-values record and the document contents are written to the LISTDSN dataset with requests separated by a high-values record.

The processing can be modified by passing a parameter to the PIPGT3B program. Several different values can be specified as program parameters. Up to three parameters are allowed, in any order (for example PARM="NDS").

Parm

Description

- D** This program parameter is used to specify whether the separators between the output parameters and document records are to be displayable. If this program parameter is specified, the separators that precede the output parameters are a line of hyphens ("-") and

those that precede the document are a line of plus signs ("+").

If this program parameter is not specified, the separator preceding the output parameters is a line of low-values, and the separator preceding the document is a line of high-values. The separators are always produced unless the program parameter "S" (single item only) is present and "M" (mixed parameters and document) is not present.

- M** This program parameter is used to specify that the output parameters and the document records are both to be put onto the LISTDSN dataset. If this program parameter is not specified the output parameters are written to CARDFILE and the document records (if any) to LISTDSN.
- N** This program parameter is used to specify that the items are not to be deleted from the outtray after they have been interpreted. If it is not present, then each item that is interpreted is deleted.
- P** This program parameter is used to specify that the document part of the interpreted item is not to be generated. Only the profile is interpreted and the output parameters are generated as usual. This parameter is only permitted when **N** is also specified.
- S** This program parameter is used to specify that only a single item on the outtray is to be processed. The first item in key sequence order is processed. All other items on the outtray are ignored.

Note: If this parameter is used, then **N** should not be used. Otherwise, only the first item on the outtray will ever be processed.

[Output Parameters](#)

The profile parameters are written to a file with one parameter per record. The parameters which may be output are described in detail in [Appendix B, "Parameters used for requests out of OfficePath"](#).

[Output Document](#)

The document dataset may be any sequential file or member of a partitioned dataset. The DCB information for LISTDSN file is examined to determine whether or not the file should contain ANSI-defined printer control characters as the first byte of each record. If the file does not have ANSI-controls then they will be removed when the text is written to the file.

[Return codes generated](#)

The return codes produced by program PIPGS3B are described in the *OfficePath Reference and Diagnosis Manual*.

Further considerations

The procedure to define a new outtray to OfficePath includes setting up a JCL record. When output is written to a Batch out-tray, the JCL record is optional (unlike the INTRDR-type destinations). If you prefer, you can omit the JCL records, in which case OfficePath will just write the requests to the out-tray and you will have to submit JCL to process the file. Sample JCL to process the file is given in [Figure 20](#) and included in member SAMPGT3B on the PIPS.JCL library. This sample JCL will print and delete the first item on the out-tray.

Note: If the interface program is called by another program you should not specify the "de-allocate on close" ("DALCLOSE" in DYNALLOC) option for the out-tray.

Figure 20. Sample JCL for running PIPGT3B

```
//JOBNAMEX JOB (ACCOUNT), 'PROGRAM',
//          MSGCLASS=A, CLASS=0, NOTIFY=XXXXXXX
//*
//*          JOB TO READ A SINGLE DOCUMENT FROM A BATCH OUTTRAY
//*
//RECEIVE  EXEC PGM=PIPGT3B, PARM='S'
//STEPLIB DD  DISP=SHR, DSN=PIPS.LOAD
//OUTTRAY DD  DISP=SHR, DSN=PIPS.VSAM.OTRAY
//SYSABEND DD  SYSOUT=*
//SYSUDUMP DD  SYSOUT=*
//SYSOUT   DD  SYSOUT=*
//CARDFILE DD  SYSOUT=*, DCB=(RECFM=FB, LRECL=133)
//LISTDSN  DD  SYSOUT=*, DCB=(RECFM=FB, LRECL=133)
//*
//ENDIT    EXEC PGM=IEFBR14
//
```

Example application

Definitions for an example application are contained in member PIPCTL4 on the PIPS.JCL library. If these have been set up using the batch control file maintenance job, PIPCTL, they can be displayed by using the online maintenance suite to view the definition for user OUTTRAYB-*, JCL member FILEB and destination FILEB.

Any mail directed to user OUTTRAYB-* will be written to the outtray defined on the destination record FILEB. OfficePath will also submit JCL record FILEB to process the outtray. This job will be submitted after the interval defined on the SYSTEM record. See the *OfficePath Installation and Administration Guide* for further details of the SYSTEM record. The JCL held on the FILEB record executes PIPGT3B to copy the document contents to DDname LISTDSN and the profile parameters to DDname CARDFILE.

Chapter 12. Outbound - using an Internal Reader

The OfficePath Batch interface can also be used to receive mail from office systems users. This allows you to construct applications which are driven by the user simply sending a document or message to a "special" userid. Alternatively, you could construct an interface to an existing mail package using the send and receive capabilities of the Batch interface. This chapter describes what definitions are necessary to allow a batch application to receive mail. It also describes how you invoke the mail interface works and how to call the interface program.

OfficePath can generate batch requests using one of two processes, either using the JES Internal reader with instream data, or by submitting JCL to process a batch outtray. This section deals with the former, and concentrates solely on outbound processing.

OfficePath processes internal reader requests by imbedding the document to be distributed within JCL that has been set up on the control file.

Several sample JCL members are supplied as standard, and these are discussed in ["The Standard OfficePath Users"](#).

How the interface operates

One or more "special" users are defined on the OfficePath control file. When mail is sent to any of these users, OfficePath intercepts the mail and creates a batch JCL deck containing the document text as instream data. This JCL deck is then submitted to JES using a CICS transient data queue which is defined as an internal reader.

The JCL deck, which is taken from the OfficePath control file, can have parameters taken for the document (for example Document name and sender name) substituted into it before the job is submitted. This means that the processing performed by the job can be controlled to a large or small extent from the user's input document.

Once the job has been submitted OfficePath has no more control over it and it is processed in exactly the same way as any other batch job.

Defining an Internal reader to OfficePath

You can have as many or as few internal readers as you wish. The DCB of the internal reader determines the DCB of the instream document text, so you will need different internal reader definitions if your application expects the document text in different formats.

Before requests are sent to an internal reader, records need to be added to the PIPCNLT file, using the on-line maintenance suite. Changes may also be required to the CICS DCT table and the CICS Start-up deck, in order to define the internal reader. The steps to be performed are listed below.

1. Define an internal reader to CICS.
2. Set up a Destination record, type of INTRDR.
3. Set up a User record.
4. Set up a JCL record.

Example definitions for carrying out steps 2 - 4 are contained in member PIPCTL5 of the PIPS.JCL library. These may have been set up when the PIPCTL job was run at installation time.

Note: During the installation process a standard internal reader called "ZRDR" should have been created. If you wish to use this then you can skip the CICS definitions.

Internal reader definition

An internal reader is used to submit jobs onto the JES input queue. It is defined to CICS by an entry in the Destination Control table (DCT) and by an entry in the CICS JCL. The DCT entry for a standard internal reader is shown in [Figure 21](#).

Figure 21. DCT table updates for an Internal reader

```

*****
*          DCT ENTRIES FOR THE SAMPLE BATCH APPLICATION          *
*                                                                 *
*****
ZAINTRDR DFHDCT TYPE=SDSCI,                                     C
          DSCNAME=ZAINTRDR, TYPEFLE=OUTPUT,                   C
          BLKSIZE=133, RECSIZE=133, RECFORM=FIXBLK
ZINT     DFHDCT TYPE=EXTRA,                                     C
          DESTID=ZINT,                                         C
          DSCNAME=ZAINTRDR,                                    C
          OPEN=INITIAL

```

For information on adding this entry, contact your CICS system programmer. Remember that DCT entries of type=EXTRA must follow all other DCT entries.

The DCT entry identifies the internal reader ZINT with the DCB information specified with record ZAINTRDR.

For every Internal reader set up in the DCT, there must be a DDname in the CICS JCL. An example definition is given below:

```

ZAINTRDR DD SYSOUT= (*, INTRDR)

```

The maximum length value which can be specified for an internal reader is 254 characters under JES2 and 80 characters under JES3. In practice, the maximum value is 133 characters since print-image documents will be truncated at this line-length by the internal processing in OfficePath.

Note: If you are using JES3 and require more than 80 characters, then you will have to use a VSAM file for the output. See ["Outbound - using a VSAM file"](#) for details.

Destination record, Internal reader type

An example of the panel used when adding a new internal reader definition to the Control File is shown in [Figure 22](#). To reach this panel, enter the on-line maintenance suite with transaction "PIPX" or the customized equivalent. From this panel, type "INSERT" as the function, "DEST" as the type and enter a key for the record that you are adding. This key can be up to 8 characters in length.

Once the program has responded, overwrite the "DESTINATION TYPE" field with "INTRDR" and press enter. This will display all of the fields that are valid for this type of record.

Once one record of the correct type is entered on the control file, it is possible to clone a new record using the existing record. This is done by displaying the existing record on the screen (using a function of "ENQUIRE" and the key of the existing record), and then overtyping the function with "INSERT" and the key with the new key. This creates a new record which can then be amended as required.

Some standard destinations should have been set up at installation time. The destinations TSO, DATASET and DCFOUT are described in more detail in ["The Standard OfficePath Users"](#).

Figure 22. The Destination panel for INTRDR type destinations

```

PIPG                               O f f i c e P a t h  V e r s i o n  1.5.0           Date: 12/01/94
Function.... ENQUIRE               Control File Maintenance           Time: 11:41:50
Type..... DEST                     Key.... READER
----- Destination Maintenance -----
      Destination type: INTRDR
DOCUMENT TYPES                      APPLICATION ATTRIBUTES
Accept RF Text..... N              Accumulate recipients..... N
Accept FF Text..... N              Can process ASA controls.... Y
Accept PC File..... N              Can process NOTE doctype.... Y
Accept Image..... N                Requires document header.... Y
Accept Editable..... Y             Returns status information... N
Accept Msg/External.. Y
Other document types.
      .....

CHARACTER SET                       CICS PARAMETERS
Accept GCID..... LOCAL
                                           Transient data queue name.... DBIN

```

```

Comments.....                               Output record format..... FBA
----- LAST MODIFIED BY ?????????? AT TERM ?????????? AT ?????????? ON ?????????? -----
PIPCFM12 Enquiry only, input Function/Type or PF key
PF 1=HELP  4=EXIT  10=PREV REC  11=NEXT REC  12=QUIT

```

Description of fields

Default values, where applicable, are indicated by brackets.

- Document Types

This set of parameters identifies the types of documents that this destination can receive. The value of the parameter is either "Y" to receive this type of document or "N" to disqualify it. If a document is sent that cannot be received by this destination, then OfficePath will attempt to convert the document to a type which the destination can accept, using a formatter program. If no suitable formatter program can be found, then the request is deleted and a **CANCELLED** status is returned to the originator of the distribution. Details of the supplied formatter programs can be found in the *OfficePath Installation and Administration* manual.

- Application Attributes

This set of attributes determines how OfficePath will format any mail that is sent to this destination. The recommended values, which are given below, are for applications which you write. If you are installing an Keane Ltd product (for example, OV/Fax), you should refer to its installation instructions for details of the settings which are required.

- Accumulate recipients

If a document is to be sent to multiple recipients on the same destination OfficePath can either deliver the request as one document with a list of recipients, or as one document per recipient. If this flag is "Y" then OfficePath will deliver the request as one document with a list of recipients. The RECIPIENT field of the profile will list all of the recipients, on this destination, to whom the document was distributed. If this flag is not set, then OfficePath will deliver multiple documents, each with one recipient.

This parameter also affects the way OfficePath uses JCL records. See ["The JCL records"](#) for details.

- Can process ASA controls

EDITABLE documents contain an ANSI-defined Print Control (ASA) character at the start of each line (for example "-", which means "advance two lines before printing"). Some user applications do not respond to these controls, resulting in bold lines (using the overstrike control "+") being processed twice. If the ASA flag is "N" then OfficePath will interpret the ANSI-defined controls (remove

overstruck lines and add blank lines) before passing the document to the application.

- Can process NOTE doctype
If an application cannot receive notes, OfficePath can alter the control information associated with a note to make it look like an EDITABLE document. With this processing, notes can be passed to applications that cannot normally cope with this type of distribution.
- Requires document header
In some instances, it is necessary to pass the document profile with the document contents. This can be achieved with the "Requires document header" When this is set, all of the profile parameters that are specified in the distribution request are passed as header information with the document. This is useful when the final destination of the document is another messaging system or a TSO dataset.
- Returns status information
If the application creates status messages (ACK requests) to indicate to the originator the outcome of the processing, then this flag should be set to "Y". OfficePath will return the status generated by the application to the originator. If this flag is set to "N", then OfficePath does not wait for the application to return a status but instead returns a status of **DELIVERED** when the request has been written to the internal reader.

Note: The **ACTIONED** (Receipt acknowledge) status must always be generated by the application, regardless of the setting of this flag.

- Line length
The line length which is used to write the data to the internal reader. If the document text on a line exceeds this value, it will be truncated. The value specified here should match the record length (LRECL) value in the DCT entry for the internal reader TDQ. If the TDQ is defined as a variable length record, then this value should be "LRECL - 4".

Note: Documents which have been processed by the formatter programs will have a maximum of 132 characters of text per line (plus 1 byte for the ASA control).

- Character Set

OfficePath can present distributed text to an application in a GCID that the application is expecting. OfficePath will convert the contents of the profile and print-image text to the character set and code page specified by the GCID value. Valid values for this are:

- "LOCAL" - Use the value from the SYSTEM record

This value should have been set to the GCID for your country during the installation process.

- "MULTI-LINGUAL" - the multi-lingual GCID defined by IBM

This is the GCID used internally by Personal Services and DISOSS.

- One of the values in [Appendix C, "Document type and Character Set Code Numbers"](#).

Note: If the document is of type RFT or FFT it will remain in multi-lingual in order to be processed by the DisplayWrite family of products.

- CICS Parameters
 - Transient data queue name

OfficePath writes all batch requests to the transient data queue identified on this panel. The name of the TDQ specified here must match up with a TDQ defined in the CICS DCT.

- Output record format

This determines the way that the document text is written to the internal reader. If the format specifies "A", then a byte will be allocated in the output queue for the text ANSI-defined (ASA) control (if the document does not have ASA controls a space will be used). Values which may be entered are:

- FB - fixed block
- VB - variable block
- VBA - variable block with ASA control characters
- FBA - fixed block with ASA control characters

Note: If the fixed/variable format setting must match the TDQ entry in the DCT. See ["Further considerations"](#) for more information about ASA controls.

User record

You must define one or more user records for every destination record. The user records can be defined explicitly or generically.

The User record panel is obtained from the main menu by typing a function of "INSERT", a type of "USER" and a unique key that will match the recipient to whom you will address the distribution. Users may be generically defined. A record with a key of BATCH-* so that all mail sent to any users with an address of "BATCH" will use the same destination. An example panel is shown in [Figure 23](#).

In order to define users explicitly, you must add user records keyed on both node and userid, separated by a hyphen. For example, BATCH-USER1 could point at READER1 and BATCH-USER2 could point at READER2.

The destination that is specified on this record must match a destination record that already exists on the control file.

Figure 23. The User maintenance panel

```
PIPA                O f f i c e P a t h  Version 1.5.0           Date: 12/01/94
Function.... ENQUIRE      Control File Maintenance       Time: 09:22:00
Type..... USER          Key.... BATCH-*
----- User Maintenance -----
 
                               Destination..... READER

Comments.....
----- LAST MODIFIED BY ZMDVKD   AT TERM WL78ZMA7 AT 09:21:57 ON 12/01/94 -----
PIPCFM12 Enquiry only, input Function/Type or PF key
PF 1=HELP  4=EXIT  10=PREV REC  11=NEXT REC  12=QUIT
```

The JCL records

When mail is sent to an internal reader, OfficePath will use the recipient userid as a key when searching for JCL on the control file. OfficePath will search for a JCL record using the following sequence, if a JCL record is not found then the program will report an error:

1. "Node-User"
2. "Node-*
3. Recipient destination

Note: If you have specified "accumulate recipients", then OfficePath will only check for JCL using the destination (since there are multiple recipients).

The JCL can be set up using the OfficePath on-line maintenance program. From the main menu (transaction "PIPX"), use a function of "INSERT", a type of "JCL", and a key. Once again, this new record can be modeled on an existing record, if one exists.

Figure 24. Example JCL record for TSO user

```
PIPC                               O f f i c e P a t h                               Date: 19/09/90
Function.... ENQUIRE              Control File Maintenance                    Time: 16:34:33
Type..... JCL                      Key.... BATCH-*
----- JCL maintenance -----
***** TOP OF DATA *****
//BATCHOUT JOB (ACCOUNT), 'BATCH', CLASS=0, MSGCLASS=A
//*
//STEP010 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD SYSOUT=*
//SYSIN DD DUMMY
//SYSUT1 DD *
&DOCUMENT_CONTENTS
//*
/* END OF JOB
***** BOTTOM OF DATA *****

----- LAST MODIFIED BY                AT TERM WU78PDA4 AT 12:22:49 ON 01/12/93 -----
PIPCFM12 Enquiry only, input Function/Type or PF key
PF 1=HELP 4=EXIT 7=BACKWARD 8=FORWARD 10=PREV REC 11=NEXT REC 12=QUIT
```

This panel allows extra lines of JCL to be inserted using the (Insert) line command "In", where "n" is an integer indicating the number of lines to be inserted. The additional lines will be placed after the current line. Individual lines can be deleted using the "D" (Delete) line command or repeated using the "R" (Repeat) line command. Each JCL record can contain a maximum of 56 lines of JCL.

Substitution parameters

It is possible to substitute information, from the document profile, into the JCL when it is submitted to the internal reader. This can be achieved by placing the name of the parameter to be substituted at the desired place in the JCL, preceded by an "&". The parameters which may be substituted are the same parameters which are output by the batch interface program (PIPGT3B), they are described in [Appendix B, "Parameters used for requests out of OfficePath"](#).

Note: All **sub-parameters** must have their names preceded by an "&" and an underscore to indicate that they are sub-parameters. For example, the "CHARS" sub-parameter would be entered in the JCL as "&_CHARS".

When OfficePath encounters a parameter name in the JCL record, it substitutes the value for that parameter from the document profile, into the JCL. However, under certain circumstances you may wish to substitute only a part of the parameter value instead of the whole. In these cases a command of the form **&PARAM(OCCURRENCE, START POS, LENGTH)** can be used.

This is similar to the substring function provided in many programming languages. The fields in the command have the following significance:

- PARAM** This is the name of the parameter from which information is to be obtained.
- OCCURRENCE** This indicates the occurrence of the parameter from which information is to be obtained. This field is specified for parameters which have multiple values.
- For example, this should always be the value 1 for the document name parameter, since a document can have only one name. On the other hand, a document may have several authors and this would allow you to select a specific author. The value defaults to
- START POS** This indicates the first character position from which information is to be obtained from the parameter value. If this is not specified it is assumed to be from the start of the parameter value, the default value is
- LENGTH** This indicates the number of characters that are to be substituted into the JCL, starting from the specified start position. The default length value is the remainder of the string from the specified start position the end of the parameter value.

The following examples are intended to give an illustration of how the command is used:

- **&RECIPIENT(1,1,8)** - this indicates that the first 8 characters from the first recipient id are to be substituted into the JCL. This results in the recipient userid being substituted into the JCL. If **&RECIPIENT** were specified then the userid and node fields would be substituted into the JCL.
- **&RECIPIENT(2,10,8)** - this indicates that 8 characters are to be substituted from the second recipient id starting from position 10. Since the recipient id is of the form, 8 userid characters followed by a space, followed by 8 characters for the node, this would substitute the node into the JCL.

If non-numeric characters are specified within the brackets, OfficePath regards this as invalid and substitutes the first occurrence of that particular parameter into the JCL and leaves the bracketed expression in the submitted JCL.

If a substitution request is made for a parameter occurrence which does not exist, OfficePath will not perform any substitution. For example, if a request is made for **&AUTHOR(2,,)** on a document which has only one author, OfficePath substitutes a blank parameter into the JCL.

Including the document text

It is possible to substitute the document text within the JCL dataset by inserting a line containing **&DOCUMENT_CONTENTS** at the position where the text is to be inserted. The "&" character must be in column 1 and the remainder of the line should be blank. If the destination is defined

with "requires document header", then all the profile parameters described above will appear before the document text.

Note: The **&DOCUMENT_CONTENTS** can only occur once in the JCL deck. Any second occurrence will be ignored. If it does not occur in the JCL deck, the document contents are included after the last line of the JCL.

If the "Output record format" includes "A" (ASA controls), then each line of document text, and each profile parameter, will include a byte for the ASA control character.

The Standard OfficePath Users

In order to provide examples of the uses of the JCL record, various special OfficePath users, and their associated JCL and destination records, are supplied with the product and their roles are described below. The OfficePath control file entries which include the sample JCL will need to be customized to your own installation standards. The sample JCL is provided on the PIPS.JCL library in member PIPCTL5, which may have been created during installation.

DATASET This entry is for a single user whose node name is the same as the default node name. When mail is sent to this user a batch job is created to copy the document text onto a sequential file. The name of the file (including member-name, if a PDS) taken from the attached message area of the send request (first 44 characters only).

The sample definitions for this user are provided on the control file with the following keys:

```
DESTINATION INTRDR
USER          PIPHOST-DATASET
JCL           PIPHOST-DATASET
```

The JCL performs the following steps:

STEP10 This step sets up an IEBFR14 on the output dataset to act as a refer back for the DCB information.

STEP20 This step uses IEBGENER to copy the document onto the output dataset (or PDS member).

STEP30 If the distribution was unsuccessful, this step is used to return a status of "ERROR" to the originator.

STEP40 If the distribution was successful, this step returns a status of

"DELIVERED" to the originator of the request.

Note: Since "DATASET" uses the attached message field, it is not suitable for the NOTE document type. This is because a NOTE cannot have an attached message.

DCFOUT This entry defines a node called "DCFOUT". Mail sent to any userid on this node will be placed in a partitioned dataset. The member name will be taken from the recipient address field.

The sample definitions for this user are provided in the following members:

```
DESTINATION INTRDR
USER          DCFOUT-*
JCL           DCFOUT-*
```

The JCL performs the following steps:

STEP10 This step uses IEBGENER to copy the document on to a Partitioned dataset.

STEP20 If the distribution was unsuccessful, this step is used to return a status of "ERROR" to the originator.

STEP30 If the distribution was successful, this step is used to return a status of "DELIVERED" to the originator.

TSO The entry defines a node called "TSO". This allows an office user to send mail to any TSO userid. The TSO userid is specified as the recipient with a node name of "TSO". The TSO userid will get a TSO message when mail arrives. He can then view the document contents using the TSO "RECEIVE" command.

The sample definitions for this user are provided in the following members:

```
DESTINATION INTRDR
USER          TSO-*
JCL           TSO-*
```

The JCL performs the following steps:

STEP10 This step uses a program that runs under TSO to send a message to a TSO user and to transmit any document contents to the TSO user. The document contents can be retrieved into a sequential dataset using the TSO "RECEIVE" command. The TSO user is supplied by the

originator of the request as the recipient name (using an address of "TSO"). The PIPMSGB program reads the MSGTEXT file and sends 64 characters from each line using command:

TSO "SEND "." USER (XXX) LOGON" command.

The "TRANSMIT" command sends the document from the dataset identified by DDname DOCTEXT to the user &RECIPIENT, on the JES system JES001 (this will need to be altered to your installation standards).

STEP20 Program PIPGS3B is used to return a "Delivered" status message to the originator if the distribution was successful.

Note: The "(,8)" is necessary to remove the node name from the OfficePath RECIPIENT parameter. See "[Substitution parameters](#)" for further details. Sending to TSO always results in a positive acknowledgement of delivery since TSO does not inform OfficePath if there have been any problems.

BATCHDSN This destination is used to copy status data (resulting from inbound requests) to a sequential file. The sequential file is identified in the inbound request by the TO DSN sub parameter of the REPLY parameter.

The sample definitions for this destination are provided in the following members:

DESTINATION BATCHDSN

JCL BATCHDSN

The JCL performs the following steps:

STEP10 This step uses IEFBR14 as a refer back of the DCB information.

STEP20 This step uses IEBGENER to copy the reply data onto the sequential dataset identified by REPLYDSN.

IMPORT This entry defines a user called "IMPORT" at the TSO node. When a message, specifying a file name, is sent to this user the file is sent to the originator of the message.

The sample definitions for this user are provided in the following members:

USER TSO-IMPORT

JCL TSO-IMPORT

The JCL performs the following steps:

- SENDDSN** This step uses PIPGS3B to send the dataset requested to the originator.
 - OK** If the distribution was successful, this step is used to return a status of "DELIVERED" to the originator.
 - ERROR** If the distribution was unsuccessful, this step is used to return a status of "ERROR" to the originator.
-

Further considerations

When sending non-print image documents out through the Internal Reader, problems can occur if the data contains X'6161' strings, that is // characters. These will be interpreted by JES as JCL cards if they occur in columns 1 and 2. This problem becomes more likely with RFT documents containing image data. OfficePath can prevent this problem by moving any non print-image document passed through the internal reader character to the right.

This function is selected using an output record format of FBA or VBA.

The internal reader interface builds JCL which is then submitted to perform the required task. Once OfficePath has submitted the job it has no way of ensuring that the job runs correctly, or at all. If you are substituting user supplied parameters, from the document profile, into the JCL, you need to ensure that no user input can cause the job to fail since the sender will be unaware of what has happened. For example, if a message "invalidmessage" is sent to the "DATASET" userid the job will fail with a JCL error because INVALIDMESSAGE is not a valid dataset name.

Sample Application

Definitions for a sample application are contained in member PIPCTL5 on the PIPS.JCL library. If these have been setup using job PIPCTL they can be displayed by using the online maintenance suite to view the definition for user BATCH-*, JCL member BATCH-* and destination READER.

Any mail directed at user BATCH-* will be substituted into the JCL BATCH-* in place of the parameter &DOCUMENT_CONTENTS. The resulting job stream is submitted to internal reader ZINT. The job uses the IEBGENER program to copy the mail to SYSOUT.

PART 3 - TSO INTERFACES

Chapter 13. Using the TSO interface

The TSO dialogs and panels that are provided with OfficePath are intended to be examples of the facilities that OfficePath can provide for the TSO user.

OfficePath allows the TSO user to participate in electronic mailing. OfficePath requires no additional pre-requisite software such as the Application Connectivity feature of OfficeVision/MVS or NetView/Access Services.

The supplied CLISTs and panels provide the following functions for the TSO user:

- To mail a dataset, or member of a Partitioned dataset, or a scripted DCF document, to any user on the Office Systems network, or to a batch or CICS application.
- To send a message to any user on the Office Systems network, or to a batch or CICS application.
- To receive status information relating to any distributions performed by the user, into a member of a PDS.
- To receive mail into a sequential dataset from any user (or CICS or Batch application) on the Office Systems network.

Note: In all cases the CLISTs and Panels are samples only, and can be tailored to meet your individual needs.

How the TSO Interface operates

Sending from TSO

The interface operates, using CLISTs and panels, from a TSO session. Each of the CLISTs can be called independently or could be accessed from the OfficePath/TSO sample menu. Whatever the function, OfficePath displays a panel that contains the mandatory fields for the distribution, together with the most commonly used optional fields.

The OfficePath CLIST validates the input parameters and writes them to a CARDFILE dataset which is passed to the batch interface program PIPGS3B in order to format the send request onto a batch in tray (default name ZPMAILBX). Any requests that are placed on this file are processed by OfficePath on a timer defined in the SYSTEM record of the Control File. All requests are routed to the appropriate OfficePath destination, depending upon the recipient(s).

Note: As the CLIST uses the batch interface program PIPGS3B, all parameters which are valid for it could be entered in the CLIST. The PIPGS3B program is described in "[Sending requests using the batch interface](#)".

[Sending to TSO](#)

Mail can be sent to a TSO user by specifying a recipient of *tsouser* and location of "TSO" on any distribution request (where *tsouser* is the recipient's TSO userid). The request is received by OfficePath, which submits JCL containing the document as in-stream data (because TSO is defined as an INTRDR type destination). The JCL invokes the TSO processor in batch and issues a TSO "TRANSMIT" command to send the document text to the TSO recipient. The OfficePath message program is also invoked to send a message to the TSO user. The "Work with Mail" CLIST uses the TSO "RECEIVE" command to obtain the mail into a sequential file which the user can edit.

[Obtaining status information](#)

Mail sent from a user is normally sent without a request for confirmation. If the user creates a PDS file to use as a status dataset (using dataset name *tsouser.PIPS.STATUS*), then requests will be sent with COD. The request specifies that the reply is to be returned to a dataset (using TO DSN) using an INTRDR type destination for the reply. When OfficePath generates the reply, it submits JCL which invokes IEBGENER to copy the in-stream COD information to the TSO user's reply dataset.

Note: This means that the CICS system requires RACF UPDATE access to the TSO user's reply dataset.

[Defining the TSO interface to OfficePath](#)

Before the TSO interface can be used, the following records must be present in the OfficePath Control file.

Note: The standard definitions referred to below may have been set up when the PIPCTL job was run during installation. These records are contained in the PIPCTL4 member on the PIPS.JCL library. After installation, the online control file maintenance transaction "PIPX" can be used to verify that these records have been setup, and to insert/amend records as required.

Intray record The example CLISTs use the standard ZPMAILBX intray to pass requests into OfficePath. The TSO interface can be amended to point to any other batch intray, so long as it has been defined according to the instructions set out in "[Defining a batch application](#)".

Note: The DD name allocated by the PIPGS3B program is always ZPMAILBX, regardless of the in-tray name within the OfficePath CICS system.

Destination records These records are used to route requests to the TSO users. Two records are required:

INTRDR Used when documents are sent to TSO users.

BATCHDSN Used to return COD to a TSO user.

User record The standard user record TSO-* points at the destination record INTRDR. Mail sent to any userid at address "TSO" causes OfficePath to look for JCL and to submit a batch job.

JCL record These records contain the JCL deck which is wrapped around any documents/COD requests which are sent to a TSO user. Two JCL records are required:

TSO-* Invoked whenever mail is sent to a TSO user.

BATCHDSN Invoked whenever COD is sent to a TSO user.

[Installing the dialogs in TSO](#)

The following procedures need to be performed in order to use the TSO interface:

1. Tailor the PIPCDFs member on the CLIST library. This member will be executed when a user first invokes the OfficePath TSO functions. This CLIST initializes the user datasets, and sets up some default values. You can edit this CLIST to ensure that the default values presented to a user are valid. See "[PIPCDFS](#)" for details of the PIPCDFs panel.
2. Add CLISTs and panels to the ISPF system. The functions required for this are:
 - a. Allocate the PIPS.CLIST library to SYSPROC
 - b. Allocate the PIPS.PANELS library to ISPLIB
 - c. Add an OfficePath option to your menu

OfficePath menu is invoked using SELECT PANEL(PIPMENU)

3. Allocate a dataset for the TSO user to receive status information.

The TSO interface assumes that the status dataset has a dataset name of "*tsouser*.PIPS.STATUS", where *tsouser* is the id of the TSO user. This dataset must be defined with the same DCB information as the internal reader used by the OfficePath BATCHDSN destination. Refer to "[Internal reader definition](#)" and "[The Standard OfficePath Users](#)" for additional information.

Note: This step is optional. If a status dataset is not found, the example CLISTS will not request status information.

Note: If you wish to "trial" the TSO CLISTs and cannot alter a logon procedure, the PIPALLOC CLIST can be used to allocate the OfficePath libraries dynamically. This CLIST will dynamically append the OfficePath libraries after your normal SYSPROC and ISPLIB dataset

concatenation. PIPALLOC **must** be run (using EXECUTE) from the TSO READY prompt every time you log on to the system, until the libraries are allocated in the usual manner. You can then use "TSO PIPMENU" from within ISPF to invoke the OfficePath menu.

The TSO dialogs

The following CLISTs and PANELs are provided with OfficePath. These are examples only, and can be customized to suit the requirements of your organization. The CLISTs and PANELs are described in more detail in the next section.

PIPMENU This CLIST displays the PIPMENU panel which presents the available functions. Selecting one of the options from this panel invokes the corresponding CLIST described below.

PIPWORK This CLIST allows the user to display the partitioned dataset that is used to hold status information.

PIPMAIL This CLIST allows the user to display the sequential dataset that is used to hold any mail that has been sent from another Office Systems user.

PIPSDOC This CLIST allows the user to send a sequential or partitioned dataset as a note or document to any user of the Office System. The CLIST displays the panel of the same name.

PIPSMSG This CLIST sends a message, keyed into the message area, to any user of the Office system. The CLIST displays the panel of the same name.

PIPCDFS This CLIST can be used to change some of the default values which are used on the other OfficePath TSO panels.

PIPSEND This CLIST is **not** invoked from the PIPMENU panel. PIPSEND is an Edit macro which can be used, while editing a file, to send a copy of the file to an Office System user.

PIPMENU

The PIPMENU CLIST is used to display the PIPMENU panel. This panel is shown in [Figure 25](#).

Figure 25. The PIPMENU panel

```
----- O F F I C E P A T H -----
SELECT OPTION ===>

 1  MAIL          Work with Mail items          OfficePath Node... TSO
                                     and Userid... TSOUSER

 2  SEND          Send a document

 3  SEND MSG     Send a message

 4  STATUS       Work with status information

 5  DEFAULTS     Change your OfficePath defaults
```

PIPWORK

This CLIST issues an EDIT command for the status dataset of the TSO user. The dataset name is unique for each user. The high level qualifier of the dataset name is the TSO userid and the second and third level qualifiers are "PIPS.STATUS". The dataset is partitioned and each member consists of a confirmation/status document for a different distribution request. The name of the member is generated by the PIPSDOC CLIST from the name of the dataset that is being sent. See "[PIPSDOC](#)".

Note: The status dataset will not receive Receipt Acknowledgements, since these are returned as normal distribution requests.

PIPSDOC

The PIPSDOC CLIST allows the TSO user to send sequential or partitioned datasets to Office systems users in either note or document format. If the format of the request is a document (either RF Text, FF Text or Editable), then the distribution can also have an attached message. The document dataset field is used to generate the name of the member to which the status information will be copied. If the dataset is partitioned then the document dataset member name is used, if the dataset is sequential then the lowest level qualifier is used as the member name.

The PIPSDOC CLIST will validate the document type, that the document dataset exists (and is a sequential file), and that at least the first SEND TO field is non-blank. It will then send the request using the OfficePath batch interface PIPGS3B.

Other parameters can be used when sending a dataset from a TSO user. Most of these are defaulted in the CLIST to enable the TSO panel to resemble the Personal Services SEND panel.

These values can be altered to suit your organization's individual requirements. A list of allowable parameters is presented in [Appendix A, "Parameters used for requests into OfficePath"](#).

The REPLY parameter is set to "Y" or "N" depending on whether the reply dataset ("*tsouser.PIPS.STATUS*") can be located. If the dataset cannot be found, then a value of "N" is used.

Note: If the Receipt ACK parameter is set to "Y", and the reply dataset cannot be found, then the Receipt ACK parameter is reset to "N".

The panel used to send a dataset is shown in [Figure 26](#).

Note: This CLIST can be invoked from the Command field of ISPF Utilities (Option 3.4) or by using "TSO PIPSDOC filename". In either case the PIPSDOC panel is displayed with the associated dataset name pre-filled.

A recipient may, optionally, be specified using "RECP(name)" in which case the CLIST will also fill in the userid (but not address) of the first recipient. For example, TSO PIPSDOC SPFLOG1.LIST RECP(PSUSR1) will send your ISPF log to PSUSR1.

[Figure 26. The PIPSDOC panel](#)

```
----- O F F I C E P A T H -----
COMMAND ==>

Document Dataset... 'PIPS.JCL(TESTNOTE) '

Document name..... NOTE CREATED IN TSO
Subject..... Demo of sending a TSO dataset using OfficePath
Author..... C M Langford                Class..... CONFIDENTIAL
                                           Doctype..... NOTE

Attached....
Message....
    .....
    .....

Send to..... PSUSER      Address.....
    .....
    .....
    .....
    .....

Personal..... N          Receipt Ack.. N

                Press ENTER to confirm or END to exit
```


PIPSMSG

The PIPMSG CLIST and panel allow the TSO user to send a message to another user of the Office system. This user can be any other user on the OfficePath system, including a Personal Services user, another TSO user, or a user on a remote node that is connected via an OfficePath SNADS interface.

The panel shown in [Figure 27](#) checks that the first message line and the first recipient is not blank before sending the distribution request into OfficePath, using the batch interface program PIPGS3B.

Note: The PIPMSG CLIST does **not** request the return of status information.

Figure 27. The PIPMSG panel

```
----- O F F I C E P A T H -----
COMMAND ==>

Message..... A message created in TSO which can be sent to any OfficePath
..... user.
..... For example, Personal Services user.
.....

Send to..... PUSER      Address.....
.....
.....
.....
.....

Press ENTER to confirm or END to exit
```

PIPCDFS

The PIPCDFS CLIST and panel allow the TSO user to change all of his default values. This panel will be invoked the first time the user invokes the OfficePath TSO panels. The default values displayed on first entry can be customized by your systems program to suit your installation standards.

The panel shown in [Figure 28](#) can also be used to create and delete the CARDFILE, STATUS and MAIL datasets which are used by the TSO interface. The mail will display "N/A" if the dataset does not exist or "DEFINED" if the dataset does exist. The dataset may be created by typing an "A" against the dataset name. Use a "D" to delete the dataset.

Note: The STATUS dataset record length must be the same as the line length of the BATCHDSN destination.

Figure 28. The PIPCDFS Panel

```
----- CHANGE YOUR OFFICEPATH DEFAULTS -----
COMMAND ==>
PRESS <ENTER> TO ACCEPT DEFAULTS
Sender's node..... TSO                and userid..... TSOUSER

Document name..... MEMO &SYSDATE &SYSTIME
Author..... John Doe                  Document class.. MEMO
                                         Doctype..... EDITABLE

Personal..... N                        Receipt Ack..... N
                                         Return Note..... N

For EDITABLE Documents
Send as doctype.. FF TEXT              Lines per page..

OfficePath Datasets
-  CARDFILE TSOUSER.PIPS.CARDFILE      DEFINED  RECFM  LREC
-  STATUS   TSOUSER.PIPS.STATUS        N/A     FB
-  MAIL     TSOUSER.PIPS.MAIL          N/A     FB

ZPMAILBX PIPS.ZPMAILBX
LOAD     PIPS.LOAD
```

PIPSEND

This CLIST is an ISPF macro that allows the PIPMENU panel to be displayed from within ISPF EDIT. It will take the name of the dataset being edited and invoke the PIPSDOC CLIST to send the dataset. A recipient may, optionally, be specified after the macro (for example PIPSEND PSUSR1). This macro gets the name of the dataset being edited and invokes the PIPSDOC CLIST.

Note: The macro does not issue a save command, and therefore the file which is sent will be the version which existed before the current edit changes.

PART 4 - CICS INTERFACES

Chapter 14. Inbound - using the CICS interface program

PIPGS3

This chapter should be read in conjunction with [Appendix A, "Parameters used for requests into OfficePath"](#).

Mail can be passed into OfficePath by utilizing a CICS application to write data to a formatted VSAM file. This file is known to OfficePath as an intray. Note that during the installation process a standard CICS intray, should have been created, known as the "Mailsack" (default DDname ZPMAILSK). This chapter describes the steps required to create another CICS intray. You can have as many, or as few, CICS intrays as you wish. Any mail that is placed on this file must be formatted in such a way that OfficePath can pick it up and distribute it successfully. To this end, Keane Ltd supply a CICS interface program that formats any data sent by the user application into intray format. This program is called PIPGS3.

The following commands are available:

- SEND Message
- SEND Note
- SEND Document
- SEND Profile
- ACK Delivered
- ACK Invalid
- ACK Cancelled
- ACK Error
- ACK Actioned

How the interface operates

The application program which interfaces to OfficePath, writes its request to a Temporary Storage queue, and LINKs to PIPGS3 passing a commarea. This commarea contains the name of the intray dataset, the name of the TSQ containing the request, and space for a four digit return code.

Each record written to the TSQ by the application program contains either a parameter name and its value, or a block of the document text. The TSQ must have the parameters first, separated from the document text records by a record containing high values in the first byte. The parameters that may be used are listed in ["Input Parameters"](#). The maximum size of a TSQ in CICS is 32767 records.

PIPGS3 uses the TSQ contents to build a record, which it then writes to the intray dataset (the name of which is given in the commarea passed from the user program).

OfficePath processes the intray files on a timer defined in the control file SYSTEM record. On this interval, OfficePath reads each intray and will process each request found thereon or NAK the request.

Defining a CICS application

A CICS application can be connected to OfficePath only when the following functions have been performed:

1. An intray file has been created, using IDCAMS.
2. An INTRAY record has been set up on the Control File.
3. A CICS FCT entry for the intray has been defined.

If you have already set up an intray, or wish to use the standard "Mailsack" intray, then you can skip the rest of this section.

Note: Status information from an inbound distribution request can be returned to an outtray. See ["Outbound - using the CICS interface program PIPGT3"](#) , for further details.

Creating the intray file

The JCL shown in [Figure 29](#) and included in PIPS.JCL member PIPITRAY can be used to define a new intray file. The JCL will need customizing to your site standards.

Figure 29. The JCL to define an INTRAY file

```
//JOBNAMEF JOB (ACCOUNT), 'PROGNAME',
//          MSGCLASS=A, CLASS=0, NOTIFY=XXXXXXX
//*****
//*
//* THIS JOB IS USED TO CREATE AND INITIALISE AN INTRAY FILE
//*
//*****
//STEP10   EXEC PGM=IDCAMS
//SYSPRINT DD  SYSOUT=*
//DD1     DD  UNIT=SYSDA, DISP=SHR,
//          VOL=SER=VOLSER      <== CHANGE TO YOUR INSTALLATION STANDARDS
//SYSIN   DD  *
          DELETE PIPS.VSAM.ITRAY FILE(DD1)
          DEFINE CLUSTER (NAME(PIPS.VSAM.ITRAY) -
                        VOL(VOLSER) -
                        FILE(DD1) -
                        FREESPACE(25 25) -
                        CYLINDERS(1 1) -
                        UNIQUE -
                        SHAREOPTIONS(2 3) -
                        DATA (NAME(PIPS.VSAM.ITRAY.DATA) -
```

```

                RECORDSIZE (2000 8192)      -
                KEYS (12 0)                 -
INDEX (NAME (PIPS.VSAM.ITRAY.INDEX))
//*
//STEP20 EXEC PGM=IDCAMS
//SYSPRINT DD  SYSOUT=*
//INPUT DD  DISP=SHR,DSN=PIPS.JCL(ZPMAIL1)
//OUTPUT DD  DISP=SHR,DSN=PIPS.VSAM.ITRAY
//SYSIN DD  *
        REPRO INFILE (INPUT)  OUTFILE (OUTPUT)
        PRINT INFILE (OUTPUT)
//

```

The intray file now needs to be defined to CICS by an entry in the File Control table. A sample definition for this file is shown in [Figure 30](#) and included in member SAMPFCT1 of the PIPS.INSTALL library.

Figure 30. Sample Intray FCT definition

```

*****
*          FCT ENTRY FOR A CICS INTRAY          *
*          (USED BY THE SAMPLE CICS APPLICATION) *
*****
DEFINE FILE (ITRAY)
        GROUP (OFFICEP)
        ADD (YES) BROWSE (YES) DELETE (YES) READ (YES) UPDATE (YES)
        RECORDFORMAT (V) STRINGS (5) RECOVERY (BACKOUTONLY)
        DSNAME (PIPS.VSAM.ITRAY)

```

An entry for this file now needs to be put in the CICS JCL. A sample definition is given below:

```

ITRAY DD DISP=SHR,DSN=PIPS.VSAM.ITRAY

```

The INTRAY record

Each intray requires an INTRAY record on the Control File so that OfficePath will process the intray.

The key of this record must match the dataset name defined in the FCT (ITRAY in our example).

Figure 31. The INTRAY maintenance panel

```
PIPB           O f f i c e P a t h   V e r s i o n 1.5.0           D a t e : 12/01/94
Function.... ENQUIRE           C o n t r o l   F i l e   M a i n t e n a n c e           T i m e : 09:25:45
Type..... INTRAY           K e y.... ITRAY
----- Input File Maintenance -----

Intray processed in batch..... N

Comments.....
----- LAST MODIFIED BY ZMDVKD   AT TERM WL78ZMA7 AT 09:25:42 ON 12/01/94 -----
PIPCFM12 Enquiry only, input Function/Type or PF key
PF 1=HELP  4=EXIT  10=PREV REC  11=NEXT REC  12=QUIT
```

The only parameter that needs to be defined for this record is whether the intray is written by a CICS, or batch (or TSO), application. This must be set to "N" for use with PIPGS3. This flag is used by OfficePath to preserve data integrity when a batch application is used.

Using the interface

The name of the TSQ that is specified in the commarea should be unique to the transaction which links to PIPGS3, and should be deleted before and after processing the request. One convention for naming the TSQ is to use the transaction ID followed by the task number (EIBTRNID, EIBTASKN).

A 20 byte commarea is passed to the PIPGS3 interface program as shown in [Figure 32](#).

Figure 32. Example LINK for PIPGS3

```
WORKING-STORAGE SECTION.

01  WS-INTERFACE           PIC X(8) VALUE 'PIPGS3  '.
01  WS-COMMAREA.
    05  WS-RETURN-CODE     PIC X(4) .
    05  WS-INTRAY-NAME    PIC X(8) .
    05  WS-TSQ-NAME       PIC X(8) .

PROCEDURE DIVISION.
```

```

EXEC CICS LINK
      PROGRAM (WS-INTERFACE)
      COMMAREA (WS-COMMAREA)
      LENGTH (20)
      END-EXEC .

```

Input Parameters

The parameters are passed to PIPGS3 on the TS queue with one parameter per TS record. The parameters which may be entered are listed below and described in detail in [Appendix A, "Parameters used for requests into OfficePath"](#).

AUTHOR The author of the document

COMMAND The OfficePath command that is issued

CORRELATE Correlation data to be returned with COD (status) information.

CLASS The document class

CREATED The date (and the time) that the document was created

DCA LEVEL A numeric code which indicates the type of document.

Sub-parameters, for Editable (print-image) documents, are:

- CHARS - number of characters per line
- LINES - number of lines per page
- OUTPUT - the required output GCID level
- FONT - the font required for the output data
- ASA - whether the output will contain ASA control characters
- MARGIN - number of characters to drop from the left of the input line

DOC DATE The date the requestor associates with the document

DOC NAME The name of the document

EXPIRES The date that the document expires

FCAB REF The location of an external reference document (44 characters maximum)

KEYWORD The keywords associated with the document

MESSAGE An attached message which is sent with the document

PASSWORD The password of the user who sent the request

PERSONAL The flag to indicate personal mail

PRIORITY The flag to indicate high priority mail

PROF GCID The GCID of the profile fields

RECIPIENT The userid and node of the recipient of the mail

REPLY Indicates whether a reply is required.

Sub-parameters, if **REPLY=Y**, are:

- TO USER - userid to whom reply information is directed
- TO SYS - destination-name of the system where replies will be directed
- TO DSN - dataset where reply data will be sent
- APPEND - attach original distribution with reply information
- RCPT ACK - whether acknowledgement of receipt is required

SUBJECT The document subject

UPDATED The date that the document was last updated

USER NAME The userid and node of the person who sent the request.

DIST CORR The distribution correlation field for use in 'receipt acknowledgement' processing.

Document

The data for a document is passed as a set of TSQ records following the parameters. The document is separated from the parameters by a record with high-values in the first byte.

For an Editable (print image) document, each record is one line of the document, including the ASA control (if present).

For any other document type, each record is whatever size the program passes, up to a maximum of

PIPGS3 Return codes

The return codes produced by program PIPGS3 are:

0000 Command successfully processed

0104 Error processing document text

Either document text has been found when processing a message, or document text was not found when it was expected. Check that the format of the request on the TSQ is correct.

0105 Internal logic error

The function passed to a sub-module was invalid. Contact technical support.

0106 Parameter sequence error

When a request is made, the **COMMAND** parameter must be first. The other parameters may appear in any order

Note: Sub-parameters must appear immediately after the relevant parameter.

0107 Parameter blank or invalid

One of the supplied parameters is invalid, or the parameter value is blank.

Note: If you want a parameter to be blank, it should not be included in the request. Only the **MESSAGE** parameter may contain blank lines.

02xx Invalid parameter data

Parameter number xx contains invalid data.

03xx Parameter invalid for request

Parameter number xx is invalid for this type of request.

04xx Sub-parameter invalid for request

Parameter number xx has a sub-parameter which is invalid for this type of request.

05xx Mandatory parameter missing

Parameter number xx is required for this type of request.

06xx Mandatory sub parameter missing

Parameter number xx requires a sub-parameter to be supplied for this type of request.

1rrr Error reading from TSQ, CICS EIBRESP is rrr.

A CICS error occurred while reading an item from the TSQ.

2rrr Error writing to in-tray, CICS EIBRESP is rrr.

A CICS error occurred while writing an item to the intray.

Some of the EIBRESP values you may receive are:

012 FILENOTFOUND

015 DUPREC

018 NOSPACE

019 NOTOPEN

044 QIDERR

xx is the number of the parameter in question, as detailed in the following list:

- 00 COMMAND
- 01 DATE-TIME
- 02 DEST-CODE
- 03 DCA LEVEL
- 04 REPLY
- 05 USER NAME
- 06 PASSWORD
- 07 QUALIFIER^(reserved for RAPID compatibility)
- 08 FROM DATE^(reserved for RAPID compatibility)
- 09 TO DATE^(reserved for RAPID compatibility)
- 10 PRIORITY
- 11 PERSONAL
- 12 PROF GCID
- 13 MESSAGE
- 14 RECIPIENT
- 15 PROCESS
- 16 CREATED
- 17 DOC DATE
- 18 EXPIRES
- 19 CLASS
- 20 DOC NAME
- 21 FCAB REF
- 22 SUBJECT
- 23 AUTHOR
- 24 KEYWORD
- 25 ACCESS
- 26 CORRELATE
- 27 UPDATED
- 28 DIST CORR

Further considerations

Avoiding processing delays

After returning from the LINK to PIPGS3, the application should check for successful completion by examining the return-code in the COMMAREA. The application can continue to write further requests to the TSQ, or can start transaction PIP5 with the following start-data. This will ensure that OfficePath will process the request immediately, rather than waiting for the interval given on the control file SYSTEM record.

- 8 bytes containing "RAPID ", padded with spaces.
- 20 bytes of low values.
- 8 bytes DDname of the intray file.

Error recovery

The interface program does not attempt to ROLLBACK any file changes that are made. The application program should perform the necessary recovery procedures.

Sample CICS inbound application

A sample CICS application is provided with OfficePath: transaction PIPT. The source for the program, PIPTTEST, the BMS source for the map, PIPTAAX, and the map copybook, PIPTAA, are available on the PIPS.SAMPLIB library. The program performs the following functions:

1. Displays a panel enabling the user to enter the details to either send a TSQ, a six line document, a six line note, or a message.
2. Receives and validates the sample panel.
 - PF2 will send the item,
 - ENTER will validate the details,
 - PF5 will pre-fill the panel with the defaults for a TSQ,
 - PF6 will pre-fill the panel with the defaults for a document,
 - PF7 will pre-fill the panel with the defaults for a note,
 - PF8 will pre-fill the panel with the defaults for a message.
3. Writes parameter information to the TSQ until all of the parameter information is written.
4. Writes a record containing high-values to the TSQ to separate the profile information from the document text.
5. Writes out the document text to the TSQ, one record for each line.
6. Writes the TSQ data to the mailsack, by linking to the PIPGS3 program.
 - If the request is for a TSQ then the TSQ is sent as a PC-FILE, otherwise a document will be sent as editable (print image).

7. Starts transaction PIP5 to avoid processing delays.
8. Deletes the TSQ used for the processing.
9. Performs a SYNCPOINT.
10. Writes an informational message to the screen, indicating that processing is complete.
11. Redisplays the panel allowing the user to send another item.

The panel displayed will look something like:

Figure 33. Example panel from the PIP5 transaction

```

PIPT                               O f f i c e P a t h                               Date: 25/05/95
Mailsack....: ZPMailsK             Test Send Facility                               Time: 10:20:34
TYPE.....: D (TsQ, Document, External, Message, or Note)
TSQ Name.....:                      or Hex...:
Document Name....:
Subject.....: INSTALLATION VERIFICATION DOC

Attached.....: THIS WAS SENT FROM A CICS SYSTEM AT RELEASE
Message.....:  3 3
                :
                :

Send To.....: KD           Address...:
Send From...: IMIUSER     Address...:

Doc Lines or Loc:
THIS IS THE FIRST LINE OF THE DOCUMENT
THIS IS THE SECOND LINE OF THE DOCUMENT
THIS IS THE THIRD LINE OF THE DOCUMENT
THIS IS THE FOURTH LINE OF THE DOCUMENT
THIS IS THE FIFTH LINE OF THE DOCUMENT
THIS IS THE LAST LINE OF THE DOCUMENT
PF2=Send, 4=Exit, 5=TSQ, 6=Document, 7=Note, 8=Message, 9=External, 10=Outtray
PRESS PF2 TO SEND

```

The document will be sent requesting a confirmation of delivery. The reply destination is "hard-coded" as **PIPTEST**. If this destination is not defined to OfficePath, the request will be processed without confirmation of delivery. An example definition for the PIPTEST destination is shown in Figure.34.

Figure 34. Example PIPTEST Destination definition

```
PIPG                               O f f i c e P a t h           Date: 11/08/93
Function.... INSERT                Control File Maintenance       Time: 15:31:06
Type..... DEST                    Key.... PIPTEST
----- Destination Maintenance -----
                Destination type: OUTTRAY

DOCUMENT TYPES                                APPLICATION ATTRIBUTES
Accept RF Text..... N                    Accumulate recipients..... N
Accept FF Text..... N                    Can process ASA controls.... N
Accept PC File..... N                    Can process NOTE doctype.... Y
Accept Image..... N                      Uses interface program..... Y
Accept Editable..... Y                   Returns status information... N
Accept Msg/External.. Y                  Outtray processed in batch... N
Other document types.                    Line-length.....
.....

CHARACTER SET                                CICS PARAMETERS
Accept GCID..... MULTI-LINGUAL           Output file name..... ZPMAILSK
                                          Mail transaction.....
                                          and terminal (optional).....

-----
PF1=Help          2=          3=          4=Exit          5=          6=
PF7=             8=          9=          10=Prev rec   11=Next rec  12=Quit
```

Chapter 15. Outbound - using the CICS interface program PIPGT3

This chapter should be read in conjunction with [Appendix B, "Parameters used for requests out of OfficePath"](#).

OfficePath can route a distribution request to an output dataset called an Outtray. Distribution requests which are routed to an outtray can then be processed by a CICS application, using the supplied interface program, PIPGT3. In addition, status information arising from an inbound distribution request can optionally be returned to an outtray for processing by PIPGT3.

How the interface operates

The application program invokes PIPGT3 by means of a standard CICS LINK command passing a commarea which contains the name of the outtray, and the name of the TSQ to be written to PIPGT3.

PIPGT3 reads the outtray and interprets the first mail item. The contents are written to the TSQ and the outtray records for that mail item are deleted.

Defining a CICS Outtray

A CICS application can receive distributions from OfficePath only when the following tasks have been performed:

1. An Outtray has been created, using IDCAMS.
2. A Destination record of type OUTTRAY has been added to the control file
3. A User record has been added to the control file.
4. A CICS FCT entry for the outtray has been defined.

Note: Example definitions to carry out steps 2 and 3 are contained in member PIPCTL4 on the PIPS.JCL library. These may have been set up at installation time when the PIPCTL job was run.

Creating an Outtray file

The JCL shown in [Figure 35](#) and included in PIPS.JCL member PIPOTRAY can be used to define a new outtray file. The JCL will need customizing to your site standards.

Figure 35. The JCL to define an Outtray file

```
//JOBNAMEX JOB (ACCOUNT), 'PROGRAM',
//          MSGCLASS=A, CLASS=0, NOTIFY=XXXXXXX
//*****
//*THIS CREATES AND INITIALISES AN OUTTRAY FILE *
//*****
//STEP10   EXEC PGM=IDCAMS
//SYSPRINT DD  SYSOUT=*
//DD1      DD  UNIT=SYSDA, DISP=SHR,
//          VOL=SER=VOLSER
//SYSIN    DD  *
           DELETE PIPS.VSAM.OTRAY FILE(DD1)
           DEFINE CLUSTER (NAME(PIPS.VSAM.OTRAY)           -
                           VOL(VOLSER)                   -
                           FILE(DD1)                     -
                           CYL(1 1)                      -
                           SHAREOPTIONS(2 3)              -
                           UNIQUE)                       -
           DATA (NAME(PIPS.VSAM.OTRAY.DATA)             -
                  RECORDSIZE(8192 8192)                 -
                  KEYS(12 0))                            -
           INDEX (NAME(PIPS.VSAM.OTRAY.INDEX) )
/*
//*
//STEP20   EXEC PGM=IDCAMS
//SYSPRINT DD  SYSOUT=*
//INPUT    DD  DISP=SHR, DSN=PIPS.JCL(ZPMAIL1)
//OUTPUT   DD  DISP=SHR, DSN=PIPS.VSAM.OTRAY
//SYSIN    DD  *
           REPRO INFILE(INPUT)  OUTFILE(OUTPUT)
           PRINT INFILE(OUTPUT)
//*
//
```

When the outtray file has been created it will need to be defined to CICS in the CICS FCT. A sample definition is given in [Figure 36](#).

Figure 36. SAMPLE FCT definition for an outtray

```
*****
*          FCT ENTRY FOR A CICS OUTTRAY *
*          (USED BY THE SAMPLE CICS APPLICATION) *
*****
DEFINE FILE(OTRAY)
  GROUP(OFFICEP)
  ADD(YES) BROWSE(YES) DELETE(YES) READ(YES) UPDATE(YES)
  RECORDFORMAT(V) STRINGS(5) RECOVERY(BACKOUTONLY)
  DSNAME(PIPS.VSAM.OTRAY)
```

An entry for this file needs to be put in the CICS JCL if a DSNAME was not included in the FCT entry. A sample entry is shown below:

Destination record, OUTTRAY type

An example of the panel used to add a new outtray definition is shown in [Figure 37](#). To reach this panel, enter the on-line maintenance suite with transaction "PIPX" or the customized equivalent. From this panel, type "INSERT" as the function, "DEST" as the type and enter a key for the record that you are adding. This key can be up to 8 characters in length, and is used only for internal processing.

The online suite allows the first character of any command to be entered in the function and type fields. The relevant program will then pad the field to the full name.

Once the program has responded, overwrite the "DESTINATION TYPE" field with "OUTTRAY" and press enter. This will display all of the fields that are valid for this type of record.

Once one record of the correct type is entered on the control file, it is possible to clone a new record using the existing one. This is done by displaying the existing record on the panel (using a function of "ENQUIRE" and the key of the existing record), and then overtyping the function with "INSERT" and the key with the new key. This creates a new record which can then be amended as required.

Figure 37. The Destination panel for OUTTRAY type destinations

```

PIPG                               O f f i c e P a t h   V e r s i o n 1.5.0           Date: 12/01/94
Function.... ENQUIRE               Control File Maintenance       Time: 11:42:49
Type..... DEST                     Key.... FILE
----- Destination Maintenance -----
      Destination type: OUTTRAY
DOCUMENT TYPES                      APPLICATION ATTRIBUTES
Accept RF Text..... N              Accumulate recipients..... N
Accept FF Text..... N              Can process ASA controls.... Y
Accept PC File..... Y              Can process NOTE doctype.... Y
Accept Image..... N                Uses interface program..... Y
Accept Editable..... Y             Returns status information... N
Accept Msg/External.. Y            Outtray processed in batch... N
Other document types.              Line-length.....
.....
CHARACTER SET                       CICS PARAMETERS
Accept GCID..... MULTI-LINGUAL     Output file name..... OTRAY
                                     Mail transaction..... MAIL
                                     and terminal (optional)..... TERM

Comments.....
----- LAST MODIFIED BY ?????????? AT TERM ?????????? AT ?????????? ON ?????????? -----
PIPCFM12 Enquiry only, input Function/Type or PF key
PF 1=HELP  4=EXIT 10=PREV REC 11=NEXT REC 12=QUIT

```

Description of fields

Default values, where applicable, are indicated by brackets.

- Document Types

This set of parameters identifies the type of documents that this destination can receive. The value of the parameter is either "Y" to receive this type or "N" to not receive this type of document. If a document is sent that cannot be received by this destination, then OfficePath will attempt to convert the document, using a formatter program, to a type which the destination can accept. If no suitable formatter program can be found, then the request is deleted and a **CANCELLED** status is returned to the originator of the distribution. Details of the supplied formatter programs can be found in the *OfficePath Installation and Administration* manual.

- Application Attributes

This set of attributes determines how OfficePath will format any mail that is sent to this destination. The recommended values, which may be given below, are for applications which you write. If you are installing an Keane Ltd product (for example, FAX), you should refer to its installation instructions for details of the settings which are required.

- Accumulate recipients

If a document is to be sent to multiple recipients, on the same destination, then OfficePath can deliver the request as one document with a list of recipients, or as one document per recipient. If this flag is "Y" then OfficePath will deliver the request as one document with a list of recipients. The RECIPIENT field of the profile will list all of the recipients, on this destination, to whom the document was distributed. If this flag is not set, then OfficePath will deliver multiple documents each with one recipient.

- Can process ASA controls

EDITABLE documents contain a Print Control (ASA) character at the start of each line (for example "-" which means advance two lines before printing). Some user applications do not respond to these controls, resulting in bold lines (using the overstrike control "+") being processed twice. If the ASA flag is "N" then OfficePath will interpret the ASA controls (remove overstruck lines and add blank lines) before passing the document to the application.

- Can process NOTE doctype

If an application cannot receive notes, OfficePath can alter the control information associated with a note to make it look like an EDITABLE document. With this processing, notes can be passed to applications that cannot normally cope with this type of distribution.

- Uses interface program

If the outtray is to be processed by either the CICS or Batch interface programs then OfficePath needs to be made aware of this. You should always set this to "Y" for any applications you write.

Note: If this flag is set to "N", OfficePath will write to the outtray using another (older) format which will not be recognized by the interface program.

- Returns status information
If the application creates status messages (ACK requests) to indicate to the originator the outcome of the processing, then this flag should be set to "Y". OfficePath will return the application supplied status to the originator. If this flag is set to "N" then OfficePath does not wait for an application generated status but returns a status of **DELIVERED** when the request has been copied onto the outtray.

Note: The **ACTIONED** (Receipt acknowledge) status must always be generated by the application, regardless of the setting of this flag.

- Outtray processed in batch
This field should be set to "N" for a CICS outtray. It is used for batch applications to ensure that file integrity is preserved between the batch and CICS processing.
- Line length
The line length of the data that is written to this file can be altered to match the line length that the application/user is expecting. If the document text on a line exceeds this value it will be truncated.

Note: Documents which have been processed by the formatter programs will have a maximum of 132 characters of text per line (plus 1 byte for the ASA control).

- Character Set

OfficePath can present distributed text to an application in a GCID that the application is expecting. OfficePath will convert the contents of the profile and print-image text to the character set and code page specified by this GCID value. Valid values for this are:

- "LOCAL" - Use the value from the SYSTEM record

This value should have been set to your country's GCID during the installation process.

- "MULTI-LINGUAL" - The IBM defined multi-lingual GCID

This is the GCID used internally by Personal Services and DISOSS.

- One of the values in [Appendix C, "Document type and Character Set Code Numbers"](#).

Note: If the document is of type RFT or FFT it will remain in multi-lingual in order that it can be processed by the DisplayWrite family of products.

- CICS Parameters
 - Output file name
OfficePath writes the results of any distribution requests for this type of destination to an output file identified in this section. The name specified here must match the name defined in the FCT.
 - Mail transaction
If a transaction is specified here, then it will be STARTed by OfficePath whenever mail/status requests are received onto the output file identified above. The transaction will be passed some start data which is described below.
 - And termid
The mail transaction specified will normally be STARTed as a background transaction. This field can be used to specify a terminal to START the transaction on.

[User record](#)

You must define one or more user records for every destination record. The user records can be defined explicitly or generically.

The User record panel is obtained from the main menu by typing a function of "INSERT", a type of "USER" and a unique key that will match the recipient to whom you will address the distribution. You can generically define a group of users such as OUTTRAY-* so that mail sent to any users who are on node "OUTTRAY" will have mail routed to the destination that this record points at. An example panel is shown below, in [Figure 38](#).

The destination that is specified in this record must match a destination record that already exists on the control file.

Figure 38. The User maintenance panel

```
PIPA           O f f i c e P a t h   V e r s i o n 1.5.0           D a t e : 12/01/94
Function....  ENQUIRE           C o n t r o l   F i l e   M a i n t e n a n c e           T i m e : 09:22:41
Type.....    USER               K e y ..... O U T T R A Y - *
-----      User Maintenance           -----

Destination..... FILE

Comments.....
----- LAST MODIFIED BY ?????????? AT TERM ?????????? AT ?????????? ON ?????????? -----
PIPCFM12 Enquiry only, input Function/Type or PF key
PF 1=HELP  4=EXIT  10=PREV REC  11=NEXT REC  12=QUIT
```

Using The Interface

The interface is invoked by an EXEC CICS LINK command passing a 20 byte commarea. The commarea contains the name of the outtray, the name of the TSQ and space for a four digit return code. An example is shown in [Figure 39](#).

It is the responsibility of the calling program to ensure that the TSQ name passed to the interface program is unique to the transaction calling PIPGT3 and is empty before LINKing. The TSQ should be deleted as soon as it is finished with. One convention for naming TSQs is to use the transaction ID followed by the task number (EIBTRNID, EIBTASKN).

Figure 39. Example LINK for PIPGT3

```
WORKING-STORAGE SECTION.

01  WS-INTERFACE           PIC X(8) VALUE 'PIPGT3'.
01  WS-COMMAREA.
    05  WS-RETURN-CODE     PIC X(4) .
    05  WS-OUT-TRAY       PIC X(8) .
    05  WS-TSQ-NAME       PIC X(8) .
```

```
PROCEDURE DIVISION.
```

```
EXEC CICS LINK  
      PROGRAM (WS-INTERFACE)  
      COMMAREA (WS-COMMAREA)  
      LENGTH (20)  
      END-EXEC.
```

Output Parameters

The parameters are written to a TSQ with one parameter per record. The parameters which may be output are listed below and are described in more detail in [Appendix B, "Parameters used for requests out of OfficePath"](#).

AUTHOR The document author(s)
CLASS The document class
CORRELATE The correlation data stored with the document
CREATED The date that the document was created
DATE-TIME The date-time key used by OfficePath for the response
DCA LEVEL The document type.

Sub-parameters with Editable (print-image) documents are:

- **CHARS** - the number of characters per line
- **LINES** - the number of lines per page of the document

DDN The unique distribution document name
DEST-CODE An internal destination code.
DIST CORR The distribution correlation data
DOC DATE The date associated with the document
DOC NAME The name of the document
EXPIRES The date that the document expires
FCAB REF The location of the external reference document
KEYWORD Keyword(s) associated with the document
MESSAGE Message text
NAK CODE The negative acknowledgement code

- **DATA** - the data accompanying a non zero NAK code

PERSONAL	Flag for personal mail
PRIORITY	Flag for high priority mail
PROF GCID	The GCID of the document profile <ul style="list-style-type: none"> • DOC GCID - the document content GCID
RECIPIENT	The qualified name (userid and node) of the user(s) to whom the document was sent
REPLY	Flag indicating if originator requested confirmation of delivery. <p>If REPLY=Y, the sub-parameters are:</p> <ul style="list-style-type: none"> • TO SYS - the destination to reply to • TO USER - the userid and node of the originator/sender of the document • APPEND - "return note with reply" flag • RCPT ACK - "Receipt acknowledgment required" flag
SENT DATE	The date and time that the document was sent
SUBJECT	The subject of the document
UPDATED	The date and time that the document was last updated
USER NAME	The userid and node of the originator of the distribution

Document contents

The document is placed on the TSQ as a series of records containing from 1 to 8190 bytes of data.

The document contents are separated from the parameters by a record containing high-values in the first byte.

For documents which are not Editable (print-image), each record contains a block of data. All characters are part of the document.

For Editable (print-image) documents, each record on the TSQ contains one line of the text. If the document contains ASA controls, then these will be in the first character position.

Note: The first character position is always reserved for an ASA control, regardless of the document type and the "ASA Controls" flag on the destination record. If the document does not contain ASA controls, then this position will contain a space character.

PIPGT3 Return codes

The program will return a non-zero return code in the commarea if an error occurs in the processing of the request.

The return codes are:

0000 Command successfully processed

0001 Application out-tray is empty

No requests were found on the specified Application Out-tray.

0100 Internal logic error

An error was returned by a sub-module. Check that the Destination record for the Out-tray on the Control File specifies 'Uses Interface program'. If not, then alter the Destination record and delete all requests from the Out-tray. If the Destination record is correct then contact the Keane Ltd Support Team as described in

1rrr Error writing to TSQ

A CICS error occurred while writing the request to the TSQ, CICS EIBRESP is rrr.

2rrr Error reading from out-tray, CICS EIBRESP is rrr.

A CICS error occurred while reading a request from the Out-tray.

3rrr Error deleting from out-tray. CICS EIBRESP is rrr.

A CICS error occurred while deleting the request from the Out-tray.

Some of the EIBRESP values you may receive are:

012 FILENOTFOUND

018 NOSPACE

019 NOTOPEN

044 QIDERR

Further Considerations

START data passed to application

If the application is STARTed by OfficePath, then it will be passed the following start data.

- 8 bytes containing "RAPID ", padded with spaces.
- 20 bytes reserved.
- 8 bytes containing the DDname of the outtray file.

Note: Even if you decide to have your transaction STARTed by OfficePath you must still allow for the possibility that all requests on the outtray may not be processed. For example, if your transaction abends, then a request will be left un-processed on the outtray. When designing your application you can allow for this by either

- Coding a start-of-day (PLTPI) transaction which checks for any unprocessed requests on the outtray.

Or

- Coding the application to process all requests on the outtray when it is started.

Error Recovery

The PIPGT3 interface program does not attempt any error recovery. It is the responsibility of the program which LINKs to it to ensure necessary recovery procedures are performed (for example, ROLLBACK). The TSQ should be deleted as it is not usually defined as a recoverable resource and would not be rolled back.

Sample CICS outbound application

A sample CICS application is provided with OfficePath: transaction PIPU. The source for the program, PIPUTEST, the BMS source for the map, PIPUAAX, and the map copybook PIPUAA are available on the PIPS.SAMPLIB library. The program performs the following functions:

1. Displays a panel allowing the user to enter an out-tray name
2. Receives the panel, checking that the out-tray exists
3. Links the PIPGT3 to "receive" the mail
4. Reads the TSQ for each parameter record, until the end of the parameters is reached. The following parameters are extracted and displayed on the screen:
 - Distribution Type - for example Message, Profile, Document
 - Send Date
 - Received date (date item written to out-tray)
 - Originator (sender)
 - The document type
 - Message (if present)
 - The first six lines, after any initial blank lines, of the document; these are only displayed for editable documents and notes
 - By pressing PF12, CEBR is invoked making it possible to view the whole of the TSQ created by the PIPGT3 program.

The program will process both mail items and confirmations of delivery; the following two figures show an example of each.

Figure 40. Example panel from the PIPU transaction (Mail Item)

```
PIPU                               O f f i c e P a t h                               Date: 11/08/93
                                      Out-tray Read                               Time: 16:21:38
Outtray....: OTRAY      Dataset.: ZMDEV.DZCICS.OTAY

Document Name....: MEMO 11/08/93 16:17:18
Subject.....: INSTALLATION VERIFICATION DOC
Mail type.....: DOCUMENT   DCA level....: 01403 - EDITABLE

Attached.....: THIS WAS SENT FROM A CICS SYSTEM AT RELEASE
Message.....:  1 7
              ....:
              ....:

Sent to.....: KDKD      -DZCICS
Received from....: IMIUSER -IMILOCN
Received.....: 11/08/1993 16:17:25  Sent.....: 11/08/1993 16:17
Document contains 0001 pages, 0006 lines      Display starts at line
THIS IS THE FIRST LINE OF THE DOCUMENT
THIS IS THE SECOND LINE OF THE DOCUMENT
THIS IS THE THIRD LINE OF THE DOCUMENT
THIS IS THE FOURTH LINE OF THE DOCUMENT
THIS IS THE FIFTH LINE OF THE DOCUMENT
THIS IS THE LAST LINE OF THE DOCUMENT
PF2=Receive, 4=Exit, 5=Delivered Ack, 6=Invalid Ack, 9=Send 12=CEBR
DOCUMENT RECEIVED VIA OFFICEPATH
```

Figure 41. Example panel from the PIPU transaction (COD)

```
PIPU                               O f f i c e P a t h           Date: 11/08/93
                                   Out-tray Read             Time: 16:17:28
Outtray....: OTRAY      Dataset.: ZMDEV.DZCICS.OTRAY

Document Name...: NAK CODE
Subject.....:
Mail type.....: COD DOC      DCA level....:      -

Attached.....:
Message.....:
      ....:
      ....:

Sent to.....: IMIUSER -IMILOCN
Received from....:
Received.....: 11/08/1993 16:17:25 Sent.....:
Document contains 0000 pages, 0001 lines      Display starts at line
RECIPIENT KDKD      DZCICS
STATUS      DELIVERED

PF2=Receive, 4=Exit, 5=Delivered Ack, 6=Invalid Ack, 9=Send 12=CEBR
DOCUMENT RECEIVED VIA OFFICEPATH
```

CICS table entries required

The CICS table entries required for this sample application (Transaction PIPU, program PIPUTEST, mapset PIPUAA) were defined when OfficePath was installed.

Any user written application program will need to be defined in the same way.

Appendix

Appendix A. Parameters used for requests into OfficePath

This chapter describes the parameters which may be used by the Batch and CICS interface programs. The parameters which may be specified, and the validation performed, is identical in both programs. The input for the Batch program is taken from a sequential file and for the CICS program from a CICS Temporary Storage queue.

The chapter contains the following sections:

- Section 1** This section introduces the command types that can be used.
 - Section 2** This section describes the command and parameter formats.
 - Section 3** This section describes each parameter which may be entered, including the restrictions on length and number of occurrences.
 - Section 4** This section provides a summary of all the commands and parameters in tabular form. The tables show the mandatory and optional parameters for each command.
 - Section 5** This section gives working examples of the various commands.
-

A.1 Input commands

Input commands to OfficePath can be divided into two sections, SEND commands and ACK commands.

SEND commands

The SEND command allows an application to distribute a message and/or document to one or more recipients. The options which are available to applications are:

- SEND MESSAGE** Send a message (up to 256 characters).
- SEND NOTE** Send a NOTE (the text must be in EDITABLE format).
- SEND DOCUMENT** Send a document (the text may be in any document type).
- SEND PROFILE** Send a document reference (no document text is allowed).

ACK commands

The ACK command allows an application, which has received a distribution, to return status information to the originator (Sender) of the request. The options which are available to applications are:

ACK DELIVERED	The mail has arrived in the recipient's in-basket (Personal Services status ARRIVED).
ACK INVALID	The recipient specified does not exist (Personal Services status INVALID).
ACK CANCELLED	The mail was a document type which the recipient could not receive (Personal Services status CANCELLED).
ACK ERROR	The mail was not delivered because an error occurred (Personal Services status ERROR).
ACK ACTIONED	The mail item has been "viewed" by the recipient (Personal Services status ACTIONED).

Note: The **ACTIONED** status should be returned by the application only if the sender requested Receipt Acknowledgement.

[A.2 Parameter format](#)

The parameter data for any request is passed into OfficePath as a series of records on either a TSQ or a file. The format of the records is the same in both cases.

Each parameter has a fixed structure as follows:

```

-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+
PARM          FIELD1    FIELD2

```

where the PARM field is 10 characters long and contains the parameter name, starting in character-position 1.

FIELD1 starts in character-position 11.

FIELD2 is optional and, if present, starts in character-position 20.

Note: If FIELD2 is not present, FIELD1 one can extend past character position 20.

The first record, of the request, must contain the command parameter as follows:

FIELD1	The value "COMMAND"
FIELD2	Either "SEND" or "ACK" as appropriate
FIELD3	The type of SEND or ACK being requested.

See COMMAND parameter below.

The other parameters can be passed in any order. Some of these parameters contain sub-parameters, for example DCA LEVEL. Where sub-parameters are used, they are identified as such by the first character of the PARM field being blank.

Note: If the input parameters for the batch interface are on a file defined as having ASA control characters (for example, FBA or VBA) then the parameters are shifted to the right one place, to allow for the ASA control character position. That is:

PARM field starts in character-position 2
FIELD1 starts in character-position 12
FIELD2 starts in character-position 21

A.3 Input parameters

AUTHOR The author(s) of the document. Multiple authors may be entered using a sub parameter of spaces. Up to 20 authors, each 20 characters long, are supported by OfficePath, any others are ignored.

Note: Personal Services normally displays only the first author entered.

AUTHOR A Person
 A N Other

Note. Personal Services does not permit the use of certain characters (including "*" and "/") to be used when specifying the author. These characters should not be used when sending to Personal Services.

COMMAND This parameter identifies the OfficePath command to be performed. The command is placed in field1 and any subcommand in field2. The valid commands are as follows:

COMMAND SEND MESSAGE
COMMAND SEND NOTE
COMMAND SEND DOCUMENT
COMMAND ACK DELIVERED
COMMAND ACK INVALID
COMMAND ACK CANCELLED
COMMAND ACK ERROR
COMMAND ACK ACTIONED

CORRELATE A field which will be returned to the originator of the SEND, in the reply ACK. This enables the originating application to match an ACK with the SEND which initiated it. Maximum length is 16 characters.

CORRELATE SEND0010

CLASS The document class, used only as documentation. Maximum length is 16

characters.

CLASS MEMO

CREATED The date (and optionally the time) the document was created. The format is dd/mm/yy hh|mm. The time may be omitted if not required. A single character may be entered if the value is less than 10.

CREATED 16/1/85 12|34

DCA LEVEL A numeric code which indicates the type of document. Valid values are from 1 to 65535.

If the document is Editable (print image), then a DCA level will be 1403 and the following sub parameters may be entered. These values are used, by the formatter programs, to create a document header when the document is converted:

- CHARS - the number of characters per line of input document

This value determines the page width in inches using the following calculation

$$\text{Page Width} = \text{CHARS} * \text{FONT-WIDTH} / 1440$$

- LINES - the number of lines per page of the input document

This value is determines the page depth in inches using the following calculation

$$\text{Page Depth} = \text{LINES} / 6$$

Note: If the document does not contain ASA controls, then this value is also used to paginate the document.

- OUTPUT - the type the document should be converted to

OfficePath supports

- R convert Editable (print image) to RFT
- F convert Editable (print image) to FFT

- FONT - the output font (x-y where x=font id, y=font width)

Used to set FONT information in document format block.

- ASA - whether or not the input contains ASA controls ("Y" or "N")

If the document contains ASA controls then OfficePath will paginate the document according to the control characters. Otherwise the document will be paginated according to the LINES parameter.

- MARGIN - number of characters to drop from left of input line

If a document contains an unwanted left margin then this parameter may be used to discard the unwanted margin. Any characters in the margin will be lost.

Note: DCA LEVEL is not required for a **NOTE**. If specified, the DCA LEVEL must be 1403.

For requests submitted from batch, this parameter may be omitted, a default DCA LEVEL of 1403 being used.

DIST CORR Distribution correlation data.

This field is used by OfficeVision products to generate RECEIPT ACKNOWLEDGEMENT. When a request is sent it is given a unique DIST CORR name. If receipt acknowledgement is requested, then the receiving application must return the DIST CORR field. The DIST CORR field generated by OfficeVision is 14 bytes long: a 14 byte hex string (characters 0-9,A-F). This field must be returned, unaltered, by the application program in the ACK ACTIONED command.

If an application is going to send a document into OfficeVision and request RECEIPT ACK then it should generate a DIST CORR value 14 bytes long in the format described above.

The field can also be used to send information from one application to another. In this case the field can be up to 36 characters, and contain any values, but it cannot be used for RECEIPT ACK.

DIST CORR C60A0A10292376

DOC DATE The date the requestor associates with the document.

Format is dd/mm/yy.

A single character may be entered if the value is less than 10.

DOC DATE 3/4/87

DOC NAME The name of the document. OfficePath supports up to 44 characters.

Note: Document name can be omitted; however Personal Services may not be able to view a document which has no name. Personal Services does not permit the use of the "*" character in the document name. This character should not be used when sending to Personal Services.

DOC NAME MINUTES OF DISOSS MEETING

EXPIRES The date the document expires. This field is for information only. Format is dd/mm/yy.

A single character may be entered if the value is less than 10.

EXPIRES 31/12/99

FCAB REF A field identifying the location of an external reference document.

OfficePath supports up to 60 characters.

FCAB REF UNDERNEATH THE ARCHES

KEYWORD Keywords which may be associated with the document. Multiple keywords may be entered by using a sub parameter of spaces. OfficePath supports up to 20 keywords, each of maximum length 60 characters, any other keywords are ignored.

Note: Personal Services does not normally display keywords.

KEYWORD KEYWORD1
KEYWORD2

MESSAGE A message which will accompany a document sent to another user.

The message is entered as 4 parameters of up to 64 characters each.

MESSAGE This is an example of a three line message. Up to 64 characters can be stored on any one line. The subparameter of the message parameter is always spaces.

PASSWORD The requestor's password.

The password may be up to 8 characters long. There is a user exit supplied with OfficePath that is invoked by the batch interface program to validate the userid and password. This module can be altered to suit your validation requirements. Refer to the *OfficePath Installation and Administration* manual.

Note: The password is ignored by the CICS interface program.

PASSWORD TOPSECRET

PERSONAL Indicates that a document is to be distributed as a personal document. Valid

values are "Y" or "N".

Note: PS/CICS does not support personal mail.

PERSONAL N

PRIORITY Indicates that a document is to be distributed as a priority document. Valid values are "Y" or "N".

Note: Personal Services does not support priority mail.

PRIORITY N

PROF GCID The GCID which the profile is in. This field identifies the character set encoding used to create the profile.

Note: If this parameter is not supplied it will default to the "LOCAL" GCID specified on the SYSTEM record on the Control File.

- DOC GCID - The GCID used to encode to document.

The format of the GCID is x-y, where x and y are between 1 and 65535.

PROF GCID 337-256
DOC GCID 337-256

RECIPIENT The recipient of the request.
For a SEND request, this is the qualified name (userid and node) of the user(s) to whom the document is being sent. More than one recipient may be entered by using a subparameter of spaces.
For an ACK request, this is the qualified name of the user to whom the status is sent (the originator of the original request). Only one recipient may be specified for an ACK.
The userid should be entered in field1 and the node in field2. The node field will default to the value on the SYSTEM record if it is not supplied.
The userid and node fields may be up to 8 characters each. OfficePath supports up to 50 recipients, any extra will be ignored.
To use a COPYLIST (defined on the Control file) enter "COPYLIST" in field2 and the copylist name in field1. In this case a copy of the document/message will be sent to all users identified in the copylist.

RECIPIENT APIUSER1 PIPHOST
MANAGERS COPYLIST

REPLY Indicates whether status information, (that is, an ACK), should be returned after processing the request, and also where, the status information should be written. Valid values are "Y" or "N". The default value is "N".
If "Y" is specified then the "reply to" details are required. If "N" is specified and the "reply to" details are supplied then a reply will be generated only if the

command fails.

The "TO DSN" and "TO USER" fields are mutually exclusive.

Note: Status information can only be returned to an application (destination type INTRDR or OUTTRAY). The status cannot be returned to a Personal Services user.

TO USER The qualified name of the reply recipient. The user should be a valid user on the destination identified by the "SYS"
The user name should be entered in field1 and the location in field2.

TO DSN The name of a sequential file (may be a member of a PDS) to which the reply is written. This name is returned in the REPLYDSN parameter (see ["Handling a confirmation document"](#)).
Up to 44 characters may be entered as the dataset name.

TO SYS The destination to which the status is written. This must correspond to a destination record on the control file.
If this value is not entered it will default to "BATCHDSN" (if TO DSN specified) or "BATCHUSR" (if TO USER specified).

APPEND This parameter causes the original NOTE to be appended to any reply sent by the recipient.
This is only valid for NOTES.

RCPT ACK This parameter is used to request confirmation that the user has viewed the mail.

The difference between receipt acknowledgement and confirmation of delivery is as follows. If you specify confirmation of delivery, confirmation is generated when the mail arrives at that user. A Receipt acknowledgement will be generated only when the user views the mail.

For example, to send the reply to a dataset specify:

```
REPLY      Y
TO DSN     STATUS.FILE(SEND1)
```

SUBJECT The subject of the document.
This field may be up to 60 characters.

```
SUBJECT    THE SUBJECT OF THE DOCUMENT
```

UPDATED The date and time the document was last updated.
The format is dd/mm/yy hh|mm.
A single character may be entered if the value is less than 10.

USER NAME The qualified name (userid and node) of the person issuing the request.
 For a SEND request, this is the name of the originator of the document.
 For an ACK request, this is the name of the user who is making the ACK (the recipient of the original request).

USER NAME APIUSER PIPHOST

A.4 Commands and parameters

This section describes the parameters that are available for the SEND and ACK commands. The key for the table is given below.

- M** Mandatory, must be specified
- O** Optional, may be specified
- Invalid, may not be specified

The SEND command

The valid parameters for each SEND command are shown below.

	MESSAGE	NOTE	DOCUMENT	PROFILE
AUTHOR	-	O	O	O
CORRELATE	O	O	O	O
CLASS	-	O	O	O
CREATED	-	O	O	O
DCA LEVEL	-	-	M	-
CHARS	-	-	O	-
LINES	-	-	O	-
OUTPUT	-	-	O	-
FONT	-	-	O	-
ASA	-	-	O	-
MARGIN	-	-	O	-
DIST CORR	O	O	O	O
DOC DATE	-	O	O	O

DOC NAME	-	0	0	0
EXPIRES	-	0	0	0
FCAB REF	-	-	-	M
KEYWORD	-	0	0	0
MESSAGE	M	-	0	0
PASSWORD	0	0	0	0
PERSONAL	0	0	0	0
PRIORITY	0	0	0	0
PROF GCID	0	0	0	0
DOC GCID	-	0	0	-
RECIPIENT	M	M	M	M
REPLY	0	0	0	0
TO USER	0	0	0	0
TO SYS	0	0	0	0
TO DSN	0	0	0	0
APPEND	0	0	0	0
RCPT ACK	0	0	0	0
SUBJECT	-	0	0	0
UPDATED	-	0	0	0
USER NAME	M	M	M	M

Note: If the REPLY parameter = "Y", then either "TO USER" or "TO DSN" must be specified.

[The Acknowledge \(ACK\) command](#)

The valid parameters for each ACK command are shown below.

	DELIVERED	INVALID	CANCELLED	ERROR	ACTIONED
CORRELATE	M	M	M	M	-
DIST CORR	-	-	-	-	M

PASSWORD	O	O	O	O	O
RECIPIENT	M	M	M	M	M
USER NAME	M	M	M	M	M

A.5 Example commands

This example shows the format of an input command. Each line on the example represents either a record on a sequential file (for Batch) or a TS record (for CICS).

The commands shown below have the same format, the first set of parameters are mandatory; the second set separated by a blank line are optional.

SEND MESSAGE command

```

PARAMETER*DATA
  SUBPARAM*FIELD1  FIELD2
----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6
COMMAND   SEND      MESSAGE
MESSAGE   THIS IS THE MESSAGE THAT WILL BE TRANSMITTED TO
          THE USER SPECIFIED IN THE RECIPIENT FIELD. THE
          MESSAGE CAN BE FOUR LINES OF 64 CHARACTERS LONG.
RECIPIENT USER2     LOCATN2
REPLY     N
USER NAME USER1     LOCATN1

CORRELATE SEND001
DIST CORR C0000000000001
PASSWORD  PASSWORD
PERSONAL  N
PRIORITY  N

```

SEND NOTE command

```

PARAMETER*DATA
  SUBPARAM*FIELD1  FIELD2
----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6
COMMAND   SEND      NOTE
RECIPIENT USER2     LOCATN2
REPLY     N
USER NAME USER1     LOCATN1

AUTHOR    A Person
CORRELATE SEND0002
DIST CORR C00000000000002
CLASS     MEMO
DOC NAME  CHRISTMAS PARTY

```

```

EXPIRES    25/12/90
KEYWORD    XMAS
           PARTY
PERSONAL   N
PRIORITY   N
SUBJECT    TEAM OUTING
UPDATED    1/1/90

```

SEND DOCUMENT command

```

PARAMETER*DATA
  SUBPARAM*FIELD1  FIELD2
-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6
COMMAND    SEND      DOCUMENT
DCA LEVEL
  ASA      N
CHARS
LINES
  OUTPUT   F
RECIPIENT  TOUSER    PIPHOST2
REPLY      N
USER NAME  FROMUSR   PIPHOST1

AUTHOR     A Person
CORRELATE  SEND003
CLASS      ACLASS
CREATED    27/09/90 12:45
DIST CORR  C0000000000003
DOC DATE   27/09/90
DOC NAME   THIS IS A TEST DOCUMENT
EXPIRES    31/12/99
KEYWORD    TEST KEYWORD
           ANOTHER KEYWORD
PASSWORD   PASSWORD
PERSONAL   Y
PRIORITY   N
PROF GCID  337-256
  DOC GCID  337-256
SUBJECT    THIS IS A TEST SUBJECT
           A N OTHER
UPDATED    27/09/90 13:30

```

SEND PROFILE command

```

PARAMETER*DATA
  SUBPARAM*FIELD1  FIELD2
-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6
COMMAND    SEND      PROFILE
FCAB REF   IN THE FILE CABINET OF ROOM
RECIPIENT  USER2     LOCATN2
REPLY      N
USER NAME  USER1     LOCATN1

```



```

AUTHOR      A Person
CORRELATE   SEND0004
CLASS       MEMO
CREATED     01/01/90
DIST CORR   C0000000000004
DOC DATE    01/01/90
EXPIRES     09/09/90
KEYWORD     DOCUMENT
            EXTERNAL
            REFERENCE
MESSAGE     THIS IS THE MESSAGE THAT WILL BE TRANSMITTED TO
            THE USER SPECIFIED IN THE RECIPIENT FIELD. THE
            MESSAGE CAN BE FOUR LINES OF 64 CHARACTERS LONG.
PASSWORD    PASSWORD
PERSONAL    N
PRIORITY    N
SUBJECT     TEST DOCUMENT
UPDATED     02/01/90

```

ACK DELIVERED command

```

PARAMETER*DATA
  SUBPARAM*FIELD1  FIELD2
-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6
COMMAND   ACK      DELIVERED
CORRELATE REPCORR001
RECIPIENT USER2    LOCATN2
USER NAME  USER1    LOCATN1

PASSWORD  PASSWORD

```

ACK INVALID command

```

PARAMETER*DATA
  SUBPARAM*FIELD1  FIELD2
-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6
COMMAND   ACK      INVALID
CORRELATE REPCORR002
RECIPIENT USER2    LOCATN2
USER NAME  USER1    LOCATN1

PASSWORD  PASSWORD

```

ACK CANCELLED command

```

PARAMETER*DATA
  SUBPARAM*FIELD1  FIELD2
-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6
COMMAND   ACK      CANCELLED
CORRELATE REPCORR003
USER NAME  USER1    LOCATN1

```

RECIPIENT USER2 LOCATN2
CORRELATE ACKNOWLEDGE CANCEL

PASSWORD PASSWORD

ACK ERROR command

```
PARAMETER*DATA
  SUBPARAM*FIELD1  FIELD2
-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6
COMMAND   ACK      ERROR
CORRELATE REPCORR004
RECIPIENT USER2    LOCATN2
USER NAME USER1    LOCATN1

PASSWORD  PASSWORD
```

ACK ACTIONED command

```
PARAMETER*DATA
  SUBPARAM*FIELD1  FIELD2
-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6
COMMAND   ACK      ACTIONED
DIST CORR C0000000000001
RECIPIENT USER2    LOCATN2
USER NAME USER1    LOCATN1

PASSWORD  PASSWORD
```

[Appendix B. Parameters used for requests out of OfficePath](#)

This chapter describes the parameters which may be generated by Batch and CICS interface programs. The parameters which may be generated are identical in both programs. The output from the Batch interface program is written to a sequential file, and from the CICS interface program to a CICS Temporary Storage queue.

The chapter contains the following sections:

- Section 1** This section introduces the types of output that can be received.
 - Section 2** This section describes the command and parameter formats.
 - Section 3** This section describes each parameter which may be received.
 - Section 4** This section provides a summary of all output types and parameters in tabular form. The table shows which parameters will appear for each type of output.
 - Section 5** This section describes how to handle a Confirmation document.
 - Section 6** This section gives examples of the various types of output.
-

[B.1 Output types](#)

Output requests from OfficePath can be divided into two types, Mail and status. The type of request can be identified from the first record.

[Received Mail](#)

When the interface program finds a mail item it returns a record which indicates the type of mail received. Types which may occur are:

- MESSAGE RECEIVED** The mail item is a message (has no document text).
- NOTE RECEIVED** The mail item is a NOTE (document text is EDITABLE).
- DOCUMENT RECEIVED** The mail item is a document (the format of document text depends on DCA LEVEL field).
- PROFILE RECEIVED** The mail item is a reference to a document (has no document text).

[Received Status](#)

When status information arrives for an application the following type will be generated.

COD DOC SEND

A confirmation of delivery document.

The document contents consist of a list of recipients and status information pertaining to a previously issued SEND command.

B.2 Parameter format

The data from the profile of the outtray item is written as a series of record to either a TSQ or file. The format of the record is similar in both scenario's. Each parameter has a fixed structure as follows:

```
-----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+  
PARM          FIELD1    FIELD2    FIELD3
```

where the PARM field is 10 characters long and contains the parameter name, starting in character-position 1.

FIELD1 starts in character-position 11.

FIELD2 is optional and if present starts in character-position 20.

FIELD3 is optional and if present starts in character-position 29.

Note: If FIELD2 or FIELD3 are not present FIELD1 can run past character-position 20. If FIELD3 is not present FIELD2 can run past character-position

The first record will always contain information which identifies the type of output request as follows:

FIELD1 The type of item read.

This will be one of the following

If the item is mail

- MESSAGE
- NOTE
- DOCUMENT
- PROFILE

If the item is status

- COD DOC

FIELD2 The mail/status identifier.

This will be

RECEIVED For mail sent to the application

SEND For status information (relating to a previous SEND).

The other parameters can appear in any order. Some of these parameters contain sub-parameters, e.g., DCA LEVEL. Where sub-parameter fields are used, they are identified as such by the first character of the PARM field being blank.

Note: There are particular circumstances in which the parameters are shifted to the right one place, to allow for an ASA control-character position.

The circumstances are:

1. When written to the internal-reader, if the INTRDR record on the Control File defines the output record format as containing ASA control characters (FBA or VBA).
2. When written to a sequential dataset by the batch interface program, if the DCB for the sequential dataset defines the file as having ASA control characters (for example, FBA or VBA).

In these circumstances:

PARM field starts in character-position 2.

FIELD1 starts in character-position 12.

FIELD2 starts in character-position 21.

FIELD3 starts in character-position 30.

B.3 Output parameters

AUTHOR The author(s) of the document.
Multiple authors are indicated using a sub-parameter of spaces. OfficePath can output up to 20 authors each of maximum length 20 characters. A document's authors will appear as

```
AUTHOR   A N AUTHOR
          A N OTHER
```

CLASS The document class.
OfficePath will output up to a maximum of 16 characters.

```
CLASS     MEMO
```

CORRELATE A field used to identify the request.
If the request is a COD DOC the correlation value will be the same value that was specified when the SEND request was made.
If the request is a mail item the correlation value should be returned, unaltered, to OfficePath by the application if it creates a status (ACK).

Maximum length will be 16 characters.

```
CORRELATE SEND0010
```

CREATED The date and time the document was created.

The format will be dd/mm/yyyy hh|mm.

```
CREATED 16/01/1985 12|34
```

DATE-TIME The date/time the request was written to the output destination.
The format will be dd/mm/yyyy hh|mm|ss.

```
DATE-TIME 06/01/1985 14|04|50
```

DCA LEVEL A numeric code which indicates the type of document. This is a five digit number in the range 00001 to 65535.
If the DCA LEVEL is 01403 then the document is in print-image (EDITABLE) format and the following sub-parameters will be given:

- **CHARS** - the number of characters per line of document.
This is the value which was specified on the destination record.
- **LINES** - the number of lines per page of the document.
This is the maximum number of lines on any page in the document.

```
DCA LEVEL 01403  
CHARS 00080  
LINES 00030
```

Note: A **NOTE** will not have a DCA LEVEL field as it is always in EDITABLE (1403) format.

DDN The unique Distribution Document Name in the following format:

8 bytes originating node address

8 bytes requestor id

4 bytes sequence number

```
DDN GBIMIC00SUPPORT 1234
```

DEST CODE A two character destination code generated internally by OfficePath to be compatible with applications designed to interface to TBS Software's RAPID product. This should not be used in any user written applications. The code is generated from characters 7 and 8 of the destination name.

```
DEST-CODE I3
```

DIST CORR Distribution correlation data.

This field contains a unique id, supplied by the originator of the request, which

can be used to exchange information between two applications or to generate receipt acknowledgment.

If this field is used to generate receipt acknowledgment it must be returned, unaltered, to OfficePath by the application on a status request (ACK ACTIONED).

The field may contain up to a maximum of 36 characters.

DIST CORR C60A0A10292376

DOC DATE The date associated with the document.
The format will be dd/mm/yyyy.

DOC DATE 03/04/1987

DOC NAME The name of the document.

OfficePath will output up to a maximum of 44 characters.

DOC NAME MINUTES OF OfficePath MEETING

EXPIRES The date the document expires. This field is for information only.
The format will be dd/mm/yyyy.

EXPIRES 31/12/1999

FCAB REF A field identifying the location of an external reference document.
OfficePath will output up to a maximum of 60 characters.

FCAB REF UNDERNEATH THE ARCHES

KEYWORD Keywords which are associated with the document. Multiple keywords will appear with a sub-parameter of spaces.
OfficePath can output up to 20 keywords each of maximum length characters.

KEYWORD KEYWORD1
KEYWORD2

MESSAGE The message text.
The message will appear as 4 lines of up to 64 characters each.

MESSAGE This is an example of a simple message
which goes over two lines.

PERSONAL Indicates if a document was distributed as a personal document.
Value may be "Y" or "N".

PERSONAL N

PRIORITY Indicates if a document was distributed as a priority document.
Value may be "Y" or "N".

PRIORITY N

PROF GCID The GCID which the profile fields are in.
The format of the GCID is xxxxx-yyyyy, where xxxxx and yyyy are between 00001 and 65535. Values which may occur are given in [Appendix C, "Document type and Character Set Code Numbers"](#).

- DOC GCID - The GCID used to encode the document.

```
PROF GCID 00337-00256
DOC GCID 00337-00256
```

RECIPIENT The qualified name (userid and node) of the user(s) to whom the document was sent.
If a document has more than one recipient these will be shown using a sub-parameter of spaces.
OfficePath can output up to a maximum of 50 recipients for one request.
The user name is shown in Field1 and the location in Field2.

```
RECIPIENT APIUSER1 PIPHOST1
          APIUSER2 PIPHOST2
```

REPLY Indicates if a reply was requested by the originator (sender) of the document/message. Value may be "Y" or "N".
The following sub-parameters will also be present:

- TO SYS - the "destination" which sent the request.
This value will always be "DISOSSPS" and is given for compatibility with RAPID applications only. This field should be ignored by user applications.
- TO USER - the qualified name of the originator (sender) of the request.
The user name will be in field1 and the node in field2.
This field can be used to generate the status (ACK).
- APPEND - Indicates if originator specified Append for a NOTE.
Value may be "Y" or "N".
Append means that the original NOTE should be appended to the reply. This is the "Return my note with reply" option in Personal Services.
- RCPT ACK - Indicates if receipt acknowledgement is required.
Value may be "Y" or "N".
Receipt acknowledgment means that a status (ACK ACTIONED) should be generated when the recipient first views the mail item.


```
REPLY      Y
  TO SYS   DISOSSPS
  TO USER  PSUSER1  PIPHOST
  APPEND   Y
  RCPT ACK Y
```

REPLYDSN The name of the dataset to which the status information should be written. This field will only occur if the original request specified "TO DSN" on the original request. The field may contain up to a maximum of 44 characters.

```
REPLYDSN  STATUS.FILE(TEST1)
```

SENT DATE The date and time the document was sent. The format will be dd/mm/yyyy hh|mm.

```
SENT DATE 16/01/1985 12|34
```

SUBJECT The subject of the document. OfficePath will output up to a maximum of 60 characters.

```
SUBJECT   THE SUBJECT OF THE DOCUMENT
```

UPDATED The date and time the document was last updated. The format will be dd/mm/yyyy hh|mm.

```
UPDATED   16/01/85 12|34
```

USER NAME The qualified name of the recipient of the distribution. This field is included for compatibility with RAPID applications; user applications should get the recipient name from the RECIPIENT field. The user name is shown in Field1 and the location in Field2.

```
USER NAME APIUSER1 PIPHOST
```

B.4 Output types and parameters

For each type of output that may appear on an outtray, the following table lists all the parameters that can appear.

Some of the parameters will appear only if they were passed to OfficePath. These are indicated by:

- M** Mandatory, value will always appear.
- O** Optional, value may appear.
- Invalid, value will never appear.

	DOCUMENT	MESSAGE	NOTE	PROFILE	COD DOC
AUTHOR	O	-	O	O	-
CLASS	O	-	O	O	-
CORRELATE	M	M	M	M	O
CREATED	M	-	M	M	-
DATE-TIME	M	M	M	M	M
DDN	M	M	M	M	O
DCA LEVEL	M	N	N	N	-
LINES	O	-	-	-	-
CHARS	O	-	-	-	-
DEST-CODE	M	M	M	M	M
DIST CORR	M	M	M	M	O
DOC DATE	M	-	M	M	-
DOC NAME	O	-	O	O	-
EXPIRES	M	-	M	M	-
FCAB REF	-	-	-	M	-

KEYWORD	O	-	O	O	-
MESSAGE	O	M	O	O	-
NAK CODE	M	M	M	M	M
DATA	O	O	O	O	O
PERSONAL	M	M	M	M	-
PRIORITY	M	M	M	M	-
PROF GCID	M	M	M	M	-
RECIPIENT	M	M	M	M	O
REPLY	M	M	M	M	-
TO SYS	M	M	M	M	-
TO USER	M	M	M	M	-
APPEND	-	-	M	-	-
RCPT ACK	M	M	M	M	-
REPLYDSN	-	-	-	-	O
SENT DATE	M	M	M	M	-
SUBJECT	O	-	O	O	-
UPDATED	M	M	M	M	-
USER NAME	M	M	M	M	-

Note: The "LINES" and "CHARS" fields will occur only for print-image (EDITABLE or 1403) documents.

The following restrictions apply to the "COD DOC"

- Either the "RECIPIENT" or the "REPLYDSN" field will occur, but not both.
- Either the "CORRELATE" or the "DIST CORR" field will occur, but not both.
- The "DDN" field will only be present if the "CORRELATE" field is.

B.5 Handling a confirmation document

A confirmation of delivery document (COD DOC) is interpreted in a similar manner to the output parameters. The COD DOC may contain status information for multiple recipients. A recipient status is returned only when the mail is delivered or deleted.

The first parameter for each recipient is always the **RECIPIENT** parameter. This will be followed by a **STATUS** sub-parameter to give the status of the distribution to that recipient.

Note: The COD DOC is generated as soon as each recipient is delivered. Therefore, if a request contains several recipients the status information may be returned at different times resulting in several COD DOC's being returned to the application (at different times). Each recipient will be confirmed once only, except for receipt acknowledgment which will result in both **DELIVERED** and **ACTIONED** state.

RECIPIENT The userid and node of the recipient. A single sub-parameter will appear to give the status of the distribution to the recipient.

```
RECIPIENT USER1    PIPHOST
STATUS     DELIVERED
```

The **STATUS** values which can occur are:

- | | |
|----------------------------|--|
| DELIVERED | The distributed item arrived in the recipient's in-basket. |
| CANCELLED | The document type could not be received by the recipient. |
| ACTIONED | The mail item has been "viewed" by the recipient. |
| **INVALID RECIPIENT | The recipient name specified does not exist. |
| **UNEXPECTED ERROR | The document was not delivered because an error occurred. |

B.6 Example output types

In the following examples, any parameters that appear before the blank line will appear in the output profile information. The parameters that appear below the blank line will only appear if they were explicitly sent with the document (the "O" parameters).

Message

```
PARAMETER*DATA
  SUBPARAM*FIELD1  FIELD2
-----+-----1-----+-----2-----+-----3-----+--
MESSAGE    RECEIVED
DATE-TIME  01/05/90 12:34:56
DEST CODE  I5
REPLY      Y
  TO SYS    DISOSSPS
  TO USER   PSUSR1   PIPHOST
RCPT ACK   N
USER NAME  USER1     LOCTN1
PRIORITY   N
PERSONAL   N
PROF GCID  00337-00256
MESSAGE    The message area
RECIPIENT  USER1     LOCTN1
SENT DATE  03/05/90 10:50
NAK CODE
```

Note

```
PARAMETER*DATA
  SUBPARAM*FIELD1  FIELD2
-----+-----1-----+-----2-----+-----3-----+--
NOTE        RECEIVED
DATE-TIME   03/05/90 12:23:45
DEST CODE   I5
REPLY       Y
  TO SYS     DISOSSPS
  TO USER    PSUSR1   PIPHOST
APPEND      Y
RCPT ACK    N
USER NAME   USER1     LOCTN1
PRIORITY    N
PERSONAL    N
PROF GCID   00337-00256
  DOC GCID   00337-00256
RECIPIENT   USER1     LOCTN1
```

CREATED 02/05/90 12:34
DOC DATE 02/05/90
EXPIRES 03/05/90
CLASS MEMO
UPDATED 04/05/90 15:34
SENT DATE 05/05/90 10:50
NAK CODE

The document text will be the contents of the NOTE in EDITABLE format

DOC NAME NOTE04/05/90
SUBJECT TEST note
AUTHOR A.J.EADE
KEYWORD Test

Document

PARAMETER*DATA
SUBPARAM*FIELD1 FIELD2
-----1-----2-----3-----+
DOCUMENT RECEIVED
DATE-TIME 06/01/1990 14:04:50
DEST CODE I2
DCA LEVEL
REPLY Y
TO SYS DISOSSPS
TO USER PSUSR1 PIPHOST
RCPT ACK N
USER NAME USER1 LOCTN1
PRIORITY N
PERSONAL N
PROF GCID 00337-00256
DOC GCID 00337-00256
MESSAGE This message accompanies a document
RECIPIENT USER1 LOCTN1
USER2 LOCTN2
CREATED 16/01/1990 12:34
DOC DATE 03/04/1990
EXPIRES 05/04/1990
CLASS MEMO
UPDATED 16/01/1990 13:34
SENT DATE 16/01/1990 15:34
NAK CODE

The document text will appear in the format determined by the DCA LEVEL

DOC NAME Minutes of the OfficePath meeting
SUBJECT The meeting to discuss OfficePath
AUTHOR A.J.EADE
KEYWORD OfficePath
TEAM

Profile

```

PARAMETER*DATA
  SUBPARAM*FIELD1  FIELD2
-----+-----1-----+-----2-----+-----3-----+--
PROFILE    RECEIVED
DATE-TIME  04/05/1990 12:23:34
DEST CODE  I3
REPLY      Y
  TO SYS    DISOSSPS
  TO USER   PSUSR1   PIPHOST
  RCPT ACK  N
USER NAME  USER1     LOCTN1
PRIORITY   N
PERSONAL   N
PROF GCID  00337-00256
MESSAGE    Attached message area
RECIPIENT  USER1     LOCTN1
CREATED    03/05/1990 12:23
DOC DATE   03/05/1990
EXPIRES    05/05/1990
CLASS      MEMO
UPDATED    03/05/1990 15:00
SENT DATE  05/05/1990 10:50
NAK CODE

DOC NAME PROFILE NAME
SUBJECT PROFILE INFORMATION
AUTHOR   A.J EADE
KEYWORD  PROFILE

```

Confirmation of delivery

```

PARAMETER*DATA
  SUBPARAM*FIELD1  FIELD2
-----+-----1-----+-----2-----+-----3-----+--
COD DOC    SEND
DATE-TIME  05/05/1990 12:23:34
DEST CODE  I4
RECIPIENT  USER2     LOCTN2
NAK CODE
DDN         LOTCN2    USER2
CORRELATE  SEND003

```

The document contents appear as shown below

```

RECIPIENT USER1   PIPHOST
           STATUS  DELIVERED
RECIPIENT USER2   PIPHOST
           STATUS  CANCELLED
RECIPIENT USER3   PIPHOST
           STATUS  **INVALID RECIPIENT
RECIPIENT USER4   PIPHOST
           STATUS  **UNEXPECTECD ERROR

```

Receipt acknowledgment

```
PARAMETER*DATA
  SUBPARAM*FIELD1  FIELD2
-----+-----1-----+-----2-----+-----3-----+-
COD DOC   SEND
DATE-TIME 05/05/1990 12:23:34
DEST CODE I4
RECIPIENT USER2    LOCTN2
NAK CODE
DIST CORR C0000000000007
```

The document contents appear as shown below

```
RECIPIENT USER1  PIPHOST
          STATUS  ACTIONED
```

Appendix C. Document type and Character Set Code Numbers

This section lists the code numbers which can be used to represent document types and character sets within OfficePath.

Character Set (GCID) codes

The following is a list of the GCID values which may be used within OfficePath. A translation table is supplied for each of these GCIDs, your installation may have defined other GCID tables which you could also use:

00337-00256	Multi-lingual (used internally by Personal Services and DISOSS)
00265-00273	Austrian / German
00684-00037	Canadian Bilingual
00281-00277	Danish / Norwegian
00313-00285	U.K English
00101-00037	U.S English
00285-00278	Finnish / Swedish
00289-00279	French
00293-00280	Italian
00305-00283	Spanish (OLD)
00309-00284	Spanish speaking

Document type (DCA) codes

The following is a list of the DCA values which may be used by an application sending/receiving requests from OfficePath using the interface programs. These values correspond to the IBM defined DCA LEVELS:

00002	Final-Form Text (FF Text)
00006	Image document
00011	Revisable-Form Text (RF Text)
00014	Personal Computer data (PC File)

00015 External document (or message)

01403 Print-image (EDITABLE)

A

- ANSI files ([26](#))
- AUTOSEND ([29](#))

C

- CICS ([44](#))

D

- Dataset
 - description of ([38](#))
- DCA LEVEL
 - descriptions ([10](#))
 - meaning ([8](#))
- DCA levels
 - list of ([51](#))
- Dcfout
 - description of ([40](#))
- DISOSS API
 - Supported requests ([5](#))
- Document Conversion ([6](#))
- Document Library Services ([1](#))
- Document types
 - See* DCA levels
- DYNALLOC ([27](#))

G

- GCID
 - meaning ([12](#))
 - on Destination record ([31](#)), ([36](#))
 - Supplied translation tables ([48](#))

I

- Import
 - description of ([42](#))
- Internal Reader ([34](#))

M

- Multiple documents ([25](#))

P

- Password Validation ([2](#))
- PIPGT3
 - description of ([46](#))

- R**
 - PIPGT3B [\(32\)](#)
 - Receipt Acknowledgment
 - Application generated [\(18\)](#)
 - RGN [\(3\)](#)
- S**
 - SAMPGS3B [\(28\)](#)
 - SAMPGT3B [\(33\)](#)
 - Status
 - Defining application to return [\(16\)](#)
 - format [\(22\)](#)
 - obtaining [\(20\)](#)
 - returning [\(24\)](#)
 - use [\(14\)](#)
- T**
 - TSO [\(43\)](#)