User guide
This user guide describes how to install and manage the Create!email software.

Operating Environment: Windows NT 4.0 or Windows 2000
Software Version: 2

May 2002
Important Copyright Notice

The information in this document is subject to change without notice and does not represent a commitment on the part of Create!form International, Inc. The software described in this document is furnished under a license or nondisclosure agreement. The software may be used or copied only in accordance with the terms of the agreement. It is against the law to copy the software onto magnetic tape, disk or any other medium for any purpose other than in accordance with the agreement. It is against the law to copy or distribute this document other than in accordance with the agreement. All rights reserved.

© Copyright 1987-2002 Create!form International, Inc. All rights reserved. Create!form, Create!print, Create!flash, Create!stream, Create!fax, Create!email and Create!archive are trademarks of Create!form International, Inc. which may be registered in certain jurisdictions. J.D. Edwards is a trademark or registered trademark of J.D. Edwards & Company in the United States and other countries. The names of all other products and services of J.D. Edwards are trademarks or registered trademarks of J.D. Edwards World Source Company in the United States and other countries. Adobe, Acrobat, Acrobat Distiller, Acrobat Exchange, Acrobat Reader and PostScript are trademarks of Adobe Systems Incorporated. All other brand/product names are trademarks or registered trademarks of their respective holders.

WARNING. This document is protected by copyright law and international treaties. Unauthorized reproduction or distribution of this document, or any portion of it, may result in civil or criminal liability.

Create!form International, Inc. will prosecute and defend its copyrights to the fullest extent of the law.

Re-order No: EM2GN-U21W
Contents

1 Introduction

1.1 Who is this Guide For? ................................. 1-2
  1.1.1 Form Design ........................................ 1-2
  1.1.2 Production Printing ................................. 1-2
1.2 User Guide Conventions ............................... 1-3
1.3 Pre-requisites ........................................... 1-4
  1.3.1 E-mail Application ................................. 1-4
1.4 Accessing this Guide ................................... 1-4
1.5 Overview of Create!email ............................. 1-5
  1.5.1 Create!form International Components .......... 1-5
1.6 Running Create!email .................................. 1-6
1.7 Recipient Details ....................................... 1-7
1.8 Installation Road Map ................................ 1-8
  1.8.1 Install Software ................................... 1-8
  1.8.2 Setup Create!email Queues ...................... 1-9
  1.8.3 Transfer Resources ............................... 1-9
  1.8.4 Provide Create!email Parameters .............. 1-10
1.9 Production Cycle ....................................... 1-11

2 Install and Configure Create!email

2.1 Create!email Requirements ............................ 2-2
  2.1.1 Requirements ..................................... 2-2
2.2 Installation Procedure ................................ 2-3
2.3 Enter License Key ..................................... 2-5
2.3.1 Review License Details ........................................ 2-6
2.3.2 Activating Create!email ....................................... 2-7
2.3.3 Starting Create!email .......................................... 2-7
2.3.4 Installing Other Components ............................. 2-7
2.4 Define a Create!email Queue ................................... 2-8
2.4.1 Add a New Printer Queue. ................................. 2-8
2.4.2 Configure Printer Queue .................................... 2-9
2.5 Configure E-mail Account Information ..................... 2-10
2.5.1 Types of Transport Protocols ....................... 2-11
2.5.2 Define Create!email Transport Protocol .......... 2-12
2.5.3 Configure E-mail Account .............................. 2-14
2.5.4 Configure Windows Identity ....................... 2-17
2.6 Create!email server Management .......................... 2-18
2.6.1 Create!email server Processing .................. 2-18
2.6.2 Working with Queued Jobs ........................ 2-19
2.6.3 The CfService Generic Service ..................... 2-20
2.7 E-mail a Test Merge Document ............................. 2-20
2.7.1 Create!email Test ......................................... 2-20
2.7.2 Merged E-mail Test ...................................... 2-22
2.8 Managing Create!email Logs .............................. 2-24
2.8.1 Log File Settings ........................................ 2-24
2.9 Unprocessed Jobs ............................................. 2-25
2.10 Failed Jobs ................................................... 2-26

3 Sending E-mails .............................................. 3-1
3.1 Overview of Job Headers ..................................... 3-2
3.2 The Create!email Job Header ............................... 3-3
3.3 E-mail Addressing ............................................. 3-4
3.3.1 Multiple Addressing on a Single Line ................ 3-6
B.2 Configure for Outlook Express ......................... B-2
  B.2.1 Configure Create!email Settings .................. B-2
  B.2.2 Configure Outlook Express Identity ............. B-3
  B.2.3 Configure the Create!email Service ............ B-3
  B.2.4 MAPI DLL’s ........................................ B-4

B.3 Configuration for Microsoft Outlook .................. B-4
  B.3.1 Configure Create!email Settings ................ B-4
  B.3.2 Configure Microsoft Outlook Identity .......... B-5
  B.3.3 Configure the Create!email Service .......... B-5
  B.3.4 MAPI DLL’s ........................................ B-6
Introduction

This chapter provides an introduction to Create!email.

The following topics are covered:

- Who is this Guide For?
- Accessing this Guide
- Overview of Create!email
- Production Cycle
- Pre-requisites
- User Guide Conventions
- Installation Road Map
1 Who is this Guide For?

1.1 Who is this Guide For?

The Create!email user guide is designed to meet the needs of persons who are responsible for the administration of the Create!email application, including the management of form projects and printer queues. The system on which you install Create!email and other Create!form International production applications is referred to throughout this guide as the e-forms server. This guide provides instructions for installing the program and for getting the most out of your e-mailed forms.

1.1.1 Form Design

To design forms to be used with Create!email, it is necessary to install the form design component (Create!form or Create!form® for JDE® OneWorld®) on a Windows PC. See the Create!form User Guide or Create!form for JDE OneWorld User Guide for further information. Your completed form resources are transferred to the e-forms server ready for production printing.

For instructions on transferring the completed form resources to the e-forms server, refer to the Create!form transfer User Guide which is installed with Create!form and Create!form for JDE OneWorld.

1.1.2 Production Printing

To automatically merge your custom forms as your spool files are printed, Create!form server or Create!form server for JDE OneWorld must be installed on your e-forms server. Once merged, your completed documents will be automatically delivered to Create!email. It is recommended that the Create!form server User Guide is readily available for reference.

The Create!send utility is provided with Create!form server. This utility facilitates the sending of spool files from your host system to the e-forms server. For more information about sending documents to Create!email, refer to the Create!form server User Guide.
1.2 User Guide Conventions

Section headings are listed at the front of each chapter to assist you in finding the relevant information. In the online version of this document, clicking on these headings will take you to the appropriate section.

The following typographical conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Convention Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bold</strong></td>
<td>Screen items and buttons.</td>
</tr>
<tr>
<td><em>Italic</em></td>
<td>Notations, references.</td>
</tr>
<tr>
<td>/G14</td>
<td>Short-cut or reference to the document, chapter or section described after the symbol.</td>
</tr>
<tr>
<td>Courier Font</td>
<td>Commands entered by a user.</td>
</tr>
<tr>
<td>SMALL CAPS</td>
<td>Keyboard keys; e.g., press SHIFT+F1 means hold down the SHIFT key and quickly press the F1 key.</td>
</tr>
<tr>
<td><img src="https://example.com" alt="Note" /></td>
<td>Additional information relating to the topic.</td>
</tr>
<tr>
<td>click on</td>
<td>Place the mouse pointer, , over the item stated and then ‘click’ (press) the left-hand mouse button once.</td>
</tr>
<tr>
<td>click and drag</td>
<td>Click on a location or object and, while pressing and holding the left mouse button, ‘drag’ the mouse pointer to a new location, then release the button.</td>
</tr>
<tr>
<td>Item&gt;Sub-Item</td>
<td>Menu selection; i.e. select the menu Item and then select the Sub-Item from the drop-down list, e.g., File&gt; New.</td>
</tr>
<tr>
<td>Numbering (1.)</td>
<td>Step-by-step procedures. To perform an action, follow these instructions in the sequential order given.</td>
</tr>
<tr>
<td>Bullet points (►)</td>
<td>Commands. To perform the action, follow the instructions given. The order is not important.</td>
</tr>
<tr>
<td>Narrow text</td>
<td>Filenames and directory locations. The name of files and directories are displayed in a different font. For example, the file New Document.txt in the Readme directory.</td>
</tr>
</tbody>
</table>
1 Pre-requisites

1.3 Pre-requisites

This user guide assumes that you are familiar with basic Windows NT/2000 concepts and terminology. You should also have an understanding of basic printer operations including the process used by the NT/2000 print spooler.

1.3.1 E-mail Application

Create!email prepares your documents for e-mailing by passing completed documents onto your nominated e-mail client or SMTP e-mail server for sending. Create!email provides support for the following mail transport protocols:

- MAPI: mail transfer protocol used by Microsoft Outlook, Exchange and Outlook Express
- VIM: mail transfer protocol used by Lotus Notes
- SMTP: (Simple Mail Transfer Protocol) protocol used by many Internet e-mail applications

You will require an account for one of these e-mail applications in order to use Create!email.

1.4 Accessing this Guide

This user guide (Create!email.pdf) is also available online in Portable Document Format (PDF).

- On the installation CD in the Manuals directory.
- From the Windows Start menu, by selecting:
  _Programs\Create!form\Help\Create!email_.
- From within the Create!email application by selecting Help\Help Topics.
1.5 Overview of Create!email

The Create!email application extends the functionality of the Create!form server software. After designing forms in the form design component, Create!form server automatically merges the custom forms with your own data files as they are printed. By adding Create!email to the application suite, your data files can be merged and e-mailed without any additional user intervention.

1.5.1 Create!form International Components

In order to use Create!email, forms are first designed in the form design component and transferred to the e-forms server. In addition, the Create!form server software must already be installed on the e-forms server.

JDE OneWorld users:

- The form design component is Create!form for JDE OneWorld which is installed on the form designer’s workstation.
- Completed form resources are transferred to the e-forms server where Create!form server for JDE OneWorld has already been installed.

All other users:

- The form design component is Create!form which is installed on the form designer’s workstation.
- Completed form resources are transferred to the e-forms server where Create!form server has already been installed.

*NOTE*  
Support for Secure PDF documents is only available when using Create!email with Create!form server (Windows) version 2.3 and above.
For the remainder of this manual:

- a reference to Create!form is also a reference to Create!form for JDE OneWorld;
- a reference to Create!form server includes Create!form server for JDE OneWorld.

The custom forms you have designed are merged with your own data using Create!form server. To e-mail the completed merged documents you will use Create!email, which is installed on the same e-forms server as Create!form server. Additional information about setting up and configuring Create!form server can be found in your Create!form server User Guide.

1.6 Running Create!email

As your host system sends spool files to the NT/2000 printer queue, Create!form server intercepts the print job and applies the required e-forms before sending the resultant documents using Create!email.

Create!email operates on a Windows NT or Windows 2000 platform, working with the print spooler. Any system that is able to print to a Windows NT/2000 print queue is able to be merged with customized e-forms.

As forms are processed through the print queue, they are automatically e-mailed with Create!email. Alternatively, documents can be automatically faxed, printed or archived using additional Create!form International modules.

The fax, print and archive facilities are sold separately according to your requirements. Contact your Create!form International distributor for more information if required.
In summary, there are four main components to the Create!email system:

1. Create!form lets you design customized e-forms for your documents.
2. Create!form server merges spool files with the e-forms you have designed.
3. Create!email processes recipient details and other information.
4. Your e-mail application sends the documents to the intended recipients.

1.7 Recipient Details

E-mail messages are sent using an e-mail address provided within the spool file. The address may be a complete Internet e-mail address, or a name that is defined in your e-mail application’s address book.

In order to resolve address book names, Create!email must be configured to use a MAPI or VIM e-mail client. This feature is not available when using an SMTP server (Section 2.5.2).

For instructions on specifying e-mail recipient details in a spool file, refer to Chapter 3, Sending E-mails.
1 Installation Road Map

1.8 Installation Road Map

This overview provides a road map on the procedures required to install Create!email as well as how to get started.

For information on installing Create!form and designing your customized form projects, refer to your Create!form User Guide.

1.8.1 Install Software

The first step is to install the required applications on the e-forms server.

---

Install Create!form server on the e-forms server. As your data files are printed, Create!form server merges your documents with your custom forms to generate the merged output. (Create!form server User Guide)

---

Install Create!email on the same e-forms server. The merged output from Create!form server can be automatically e-mailed using your existing MAPI, VIM, or SMTP mail application. The merged document is sent as a PDF file attachment to the recipient identified in the data file. (Chapter 2)

---

(Optional) To also enable your data files to be split, redirected, or copied as they are printed, install Create!stream on the e-forms server. (Create!stream User Guide)

---
1.8.2 Setup Create!email Queues

To enable print jobs to be managed by Create!email, a printer queue must be created on the e-forms server.

Create a standard Windows printer queue that outputs to the CpEmailPort printer port.

(\textit{\textcopyright} Section 2.4)

Identify the printer queue as a Create!form server queue using the default properties.

(\textit{\textcopyright} Create!form server User Guide)

1.8.3 Transfer Resources

In order to put your custom forms into production, form projects and other resources must be transferred to the e-forms server.

Form projects and required resources can be transferred to the e-forms server using the Create!form transfer module.

(\textit{\textcopyright} Create!form transfer User Guide)
1 Installation Road Map

1.8.4 Provide Create!email Parameters

To specify the form project required for merging and other job details, parameters are sent to Create!form server along with the print job. Create!form server and Create!email support a variety of job parameters that can be included with the job.

1. Identify the parameters you will want to include with your Create!email jobs (Chapter 3).

2. Attach the parameters to the spool file document and send the spool file with attached header to Create!form server.

   - It is recommended that Create!stream is used to attach the required parameters to the spool file document before sending the job to Create!form server (Create!stream User Guide).

   Create!stream lets you fully configure your e-forms output for distributed splitting of data files into individual documents before merging. You can also obtain e-mail addresses automatically from a database or from the data file document itself and provide job parameters ‘on the fly’. See your Create!form International distributor for more details about Create!stream.

Additional methods can be used to include some parameters (see Appendix A for a detailed listing of all available parameters). Please refer to the Create!form server User Guide for details.
1.9 Production Cycle

When you print from your host application, Create!form server produces the merged output as a PDF file. Create!email then sends the resultant PDF file to the nominated e-mail address as an attachment.

The following diagram depicts the basic flow of a data file document as it enters the Create!form production process:

![Diagram of Production Process](image)

*Figure 1-1: Production process used to generate Create!email output*
This chapter describes how to install Create!email on the e-forms server and configure Create!email for use with your e-mail client.

The following topics are covered:

- **Create!email Requirements**
- **Define a Create!email Queue**
- **Create!email server Management**
- **Managing Create!email Logs**
- **Installation Procedure**
- **Configure E-mail Account Information**
- **E-mail a Test Merge Document**

Installation and configuration of Create!form server is explained in the **Create!form server User Guide**.
2 Create!email Requirements

2.1 Create!email Requirements

The Create!email installation CD-ROM contains the Create!email server application which is installed on the e-forms server. This machine must be the same machine that Create!form server has been installed on. You will require Administrator privileges in order to perform the installation.

2.1.1 Requirements

The following is required to install and use Create!email:

- Create!email installation CD-ROM.
- Microsoft Windows NT version 4.0 or Microsoft Windows 2000.
- Approximately 10 Mb free hard disk space.
- 32 Mb RAM.
- Create!form server 2.3 or greater is required to send secure PDFs via e-mail.
- Adobe Acrobat Distiller version 4.0 or greater, which is a component of the full Adobe Acrobat suite.

Acrobat Distiller is not required when installing Create!email for use with Create!form server for JDE OneWorld.

Additional software requirements:

- Create!form server must already be installed on the e-forms server before you can install Create!email (Create!form server User Guide).
- A full client installation of Lotus Notes or Microsoft Outlook/Exchange is required on the e-forms server where Create!email server is installed; alternatively, you can access a remote e-mail server using the SMTP protocol (Section 2.5 for additional information on configuring Create!email to work with your e-mail client).
Optional components:

- To enable Create!email job parameters to be easily added to your data files, it is also recommended that Create!stream server is installed on the e-forms server.

Upgrading Create!email:

- When upgrading an existing Create!email installation, the setup procedure will require you to stop the print spooling service. It is recommended that you do not have production print jobs running while you are performing the upgrade.

2.2 Installation Procedure

To install Create!email on the e-forms server:

1. Log on to the Windows e-forms server as a user with 'Administrator' rights.
2. Insert the Create!email installation CD-ROM into the CD-ROM drive. The Create!email setup screen should be displayed automatically.

   \[\textit{If the Create!email setup screen is not started automatically, run the CEmail.exe program located in the root directory of the CD-ROM.}\]

3. Select \textbf{Install Products}, then choose \textbf{Create!email}.
4. Setup will automatically install Create!email to the same root directory used for installing Create!form server (the default installation directory is \texttt{C:\Program Files\Create!form}).
2 Installation Procedure

5. Continue through the installation program by pressing Next, and follow the on-screen instructions. By default, Setup installs support for MAPI, VIM and SMTP protocols.

In order to install Create!email support for Lotus Notes, a local Lotus Notes client must already be installed on the e-forms server. If you have not yet installed a Lotus Notes e-mail client, you may Cancel Setup and perform the Create!email installation after you have installed Lotus Notes.

6. If you are upgrading Create!email, a port processor has already been installed. In order to upgrade the port processor, Setup needs to stop the print spooler service.
   - If you have previously installed Create!email on this system, select Yes to stop the spooler service. This may affect print jobs currently being queued or printed.
   - If this is the first time you have installed Create!email on this system, there is no need to stop the print spooler service.

7. Once installation is complete, you will return to the Create!email setup screen. From the Main Menu, click Exit to finish, or continue installing additional products if required.
2.3 Enter License Key

Before you can use Create!email in production, you must obtain a license key from your distributor. You may be given a temporary license key if you are evaluating the product, however, for continued operation you will require a permanent license key. In order to issue you with a permanent license key, you must inform the distributor of your Authorization code.

To view the Authorization Code:

1. Start Create!email if it is not already started:
   - Select Programs>Create!form>Create!email from the Start menu.

2. From the Main menu, select Administration>License Keys.
   The Create!form license key dialog will be displayed, with the Authorization code at the top of the dialog.

3. Once you have notified your distributor of the Authorization code, you will be given a license key.
2 Enter License Key

To enter the Create!email license key:

1. Select **Administration Licenses**... from the Main menu.
   The Create!form license key dialog will be displayed, detailing any existing
   *Create!form International* licenses.
2. Click **Create** to enter a new license key.
   The Create license key dialog will be displayed.
3. Type the license key you have been given and click **OK**.

---

**NOTE**

*A permanent product license will require one license key per product. Several products may be enabled with the one license key if you have been issued with a temporary or evaluation license.*

2.3.1 Review License Details

To check that the license key information is correct and current, you can review the license information in the Create!form license key dialog.

To review license details:

1. Select **Administration Licenses**... from the Main menu.
   The Create!form license key dialog will be displayed, detailing any existing
   *Create!form International* licenses.
2. Select the license key to review and click **Display**.
   The products enabled by the selected license key are displayed, along with the license expiry date and licensed number of printers if appropriate.
3. Click **OK** to close the dialog, then click **Exit** to return to the Create!email server application.
2.3.2 Activating Create!email

Create!email operates as a Windows service which is automatically started whenever your system is powered on. If for some reason, the service is stopped during a Windows session, it will be automatically started again when the Create!email server application is started. The service remains active as long as the e-forms server is switched on. However, for security purposes you can log off the e-forms server without affecting Create!email processing.

Once a printer queue on the Windows e-forms server has been established using the CpEmailPort output port (Section 2.4), any jobs sent to that printer queue automatically activate the Create!email service.

2.3.3 Starting Create!email

Once installation has been completed, the CfService (that Create!email utilizes) will be started automatically. To start the Create!email server application, select:

- Programs\Create!form\Create!email from the Start menu.

Close the Create!email server window by clicking the Close icon ( ) in the top right corner of the application’s window.

The CfService continues to run as long as the e-forms server is switched on.

2.3.4 Installing Other Components

The Create!email installation CD-ROM contains additional components for working with Create!form International products.

Acrobat Reader is required in order to view the Create!email and other Help files.

- If necessary, you can install Acrobat Reader from the Create!email installation screen (Section 2.2).

- Alternatively, you can install Acrobat Reader from the \Utilities\Acrobat Reader directory on the installation CD-ROM.
2 Define a Create!email Queue

2.4 Define a Create!email Queue

Create!email works in conjunction with a Create!form server printer queue, automatically merging, then e-mailing jobs as they are received. To configure your e-forms server for use with Create!email requires three simple steps:

- A Windows printer queue must be available for Create!email jobs. If necessary, use the Add Printer function to create a new printer that uses a Generic/Text Only printer driver (Section 2.4.1).

- Create!email automatically processes all jobs that are sent to the printer output port called CpEmailPort. In the Windows printer properties, select CpEmailPort as the output port for the Create!email queue (Section 2.4.2); this port was made available when you installed Create!email.

- To enable merging of custom forms, use Create!form server to configure the queue with the appropriate print processor. Use the default Create!form server properties when configuring a queue for Create!email (Section 2.4.2).

Once a printer queue has been setup for Create!email as described above, any jobs sent to that printer queue will automatically activate the Create!form server and Create!email services. Create!form server processes the job and delivers it to the CpEmailPort output port specified on the printer queue.

---

Create!form server for JDE OneWorld must be running in order to process print jobs as they are sent. However, Create!form server (Windows) will be automatically launched when a print job is received, even if it is not running.

---

2.4.1 Add a New Printer Queue

If necessary, you will need to define a new printer queue using the Add Printer function in the Windows Printers applet (refer to the Microsoft Windows User Guide for more information on adding a printer).
2.4.2 Configure Printer Queue

The printer queue on the e-forms server is required for managing the output of all merged documents. The printer queue determines the print processor that will be used, as well as the output port for sending completed documents. The output can be directed to a physical printer, or may be distributed as e-mail, fax or archive through an appropriate port. To configure a printer queue for use with Create!email, the following steps must be taken:

► The printer driver you must use for the printer queue depends upon the output port the job will be directed to. For handling Create!email jobs, the printer queue should be configured with a **Generic/Text Only** printer driver.

► In order to automatically merge and e-mail incoming documents as they are printed, the queue you have defined must be configured by Create!form server. This will activate Create!form server as jobs are sent to that queue.

► To deliver merged jobs to the Create!email service for processing, the printer queue must use the **CpEmailPort** as its output destination (see below).

**Attach the CpEmailPort to the printer queue:**

Installation of Create!email also installs the Create!email port processor called **CpEmailPort**. It is necessary that this port is attached to the printer queue that Create!email jobs will be sent to.

---

**NOTE**

*Before making changes to the print processor or port for a printer queue, it is a good idea to check that no jobs are currently being queued or printed.*

---

1. Using the Windows printer commands, display the properties of the printer queue you wish to use for Create!email.
2 Configure E-mail Account Information

2. Select **CpEmailPort** as the output port for the printer.

![Table showing Printer Ports](image)

*Figure 2-2: Select CpEmailPort as the output port for the printer queue*

Any jobs sent to a printer queue using **CpEmailPort** will be automatically processed by the Create!email application.

**Configure queue with Create!form server:**

The Create!form server application is used to configure the defined printer queue for merging, by attaching the Create!form server print processor to the queue. Either a new or existing printer queue can be configured as a Create!form server queue.

1. If necessary, create a new printer queue first ([Section 2.4.1](#)).
2. Refer to the **Create!form server User Guide** for instructions on configuring a Create!form server queue.

When defining a queue for use with Create!email, there is no need to alter any of the default Create!form server properties. However, to modify form project directories or other settings, consult the relevant section of the **Create!form server User Guide** for details.

### 2.5 Configure E-mail Account Information

To complete the Create!email configuration, the application requires information about your e-mail protocol and how to log on to your account. These topics are discussed in this section.

*NOTE: Whenever changes are made to Create!email’s configuration, the Create!email server queue is automatically paused. Any jobs held on the queue at this time will be delivered using the new configuration once the queue is restarted ([Section 2.6](#)).*
2.5.1 Types of Transport Protocols

Before commencing configuration, determine which method you will use to access your e-mail server:

1. Access your e-mail server through a local e-mail client:

Access your Lotus Notes or Microsoft Outlook/Exchange e-mail account using the full client installation of your e-mail software:

- Configure Create!email to use the VIM (Lotus Notes) or MAPI (Microsoft Outlook/Exchange) transport protocol as appropriate.
- Access to your e-mail account through your e-mail client provides full Create!email functionality, however, you will be unable to modify the name of the sender (the FROM value) in e-mails delivered by Create!email.

Note: The appropriate e-mail client must be installed on the e-forms server prior to using Create!email.

2. Access your e-mail server directly:

Access your SMTP-capable e-mail server software directly using Simple Mail Transfer Protocol (SMTP). Wherever possible, it is recommended that SMTP is used to access your e-mail account as this provides greater flexibility, although there are some limitations:

- Accessing an e-mail account using the SMTP protocol allows you to dynamically assign the sender's name to sent e-mail, but you will be unable to assign sent e-mails to a specific SENT folder on your server. To monitor those e-mails successfully sent to the SMTP server refer to the Success log file, Section 2.8.
- Create!email cannot lookup an address book name when the SMTP protocol is used. The full e-mail address must be provided to Create!email.
2 Configure E-mail Account Information

2.5.2 Define Create!email Transport Protocol

To allow Create!email to deliver e-mail messages using your e-mail account, you will need to specify the type of mail protocol you are using, and provide log off details.

Select the e-mail transport protocol:

1. Start Create!email server if it is not already started.
2. Select **Administration**→**Configure email** from the Main menu. The Email Settings dialog will be displayed.

![Email Settings dialog](image)

Figure 2-3: Configure the Create!email transport protocol

**NOTE**

**VIM (Lotus Notes)** email transport will not be available unless support for Lotus Notes was installed ([Section 2.2](#)).

3. In the **Email transport** section, select the email protocol you will use for sending e-mails generated by Create!email. Refer to [Section 2.5.1](#) for information on choosing the appropriate protocol.
Miscellaneous settings:

When a job is sent to a Create!email printer queue, Create!email will automatically logon to your e-mail application if it is not already running. In the Miscellaneous settings section, you can specify when Create!email should logoff the e-mail account. You can also specify an e-mail address to send notifications to should a failure occur.

4. In the Logoff account after field, enter the number of minutes that Create!email should wait before logging off the e-mail account.
   - Create!email will logon to the e-mail client when it receives a job, and will log off when the specified time elapses without having received further jobs. If further jobs are received prior to the logoff interval, the timer will be reset.
   - Enter a value of 0 (zero) to cause Create!email to logon to your e-mail client when the Create!email service is started (Section 2.6), and only logoff again if the Create!email service is stopped.

5. Create!email handles incomplete e-mail messages by sending an error notification to a nominated e-mail address. An incomplete message is one that has a missing TO parameter (Section 3.3), or contains a command that is unable to be carried out, such as a command to attach a file that cannot be found (Section 3.6).

Enter the e-mail address to send failure notifications to in the On failures send email to field.
   - When an incomplete message is received, an e-mail will be sent to the nominated e-mail address, such as a system administrator.
   - When using VIM or MAPI protocols, a copy of the notification will also be stored in the Sent folder of the e-mail account in use. No other e-mail is sent.
   - The e-mail error notification will contain a description of the error, the offending header and the original data file as an attachment:

    ![Sending Email](image)

    *Figure 2-4: A failed e-mail message sent to an administrator*
2 Configure E-mail Account Information

6. If the **On failures send email to** field is left blank, error notifications are not sent, but an entry will be recorded in the Create!email log file (*Section 2.8*). The original message will never reach the e-mail software for sending.

   Regardless of the method chosen for error handling, Create!email will record all activities in the Trace log file (*Section 2.8*).

7. To continue configuring Create!email for your selected email transport, click the **Configure** button in the Email Settings dialog. The next section provides details of how to configure an account for each transport protocol.

### 2.5.3 Configure E-mail Account

In order to define a specific account for use with Create!email jobs, a new e-mail account must be created on your e-mail server. Once available, you can specify how Create!email should logon to the appropriate e-mail account. The method you use will depend on the Email transport you have selected in the Email Settings dialog.

**Configure SMTP log-on:**

1. After choosing SMTP as the Email transport, click the **Configure** button to display the SMTP Configuration options.

2. Enter the TCP/IP address of the SMTP e-mail server in the **SMTP server name** field. If your system supports it, you may also identify your SMTP server using its domain name.

3. If your SMTP server requires logon authentication, check the **This server requires logon** check box and enter the details in the **User name** and **Password** fields.

4. To ensure that Create!email is able to access your SMTP e-mail server, click the **Test** button. A dialog will confirm whether or not the test was successful. Create!email will attempt to contact your mail server for up to 60 seconds. If after that time, it has not received a response, the test will fail.

   If the test fails, check that you have entered the correct IP address and user logon details if required. If the test is successful, click **OK** to resume configuring your e-mail account.
5. An SMTP mail server that supports extended SMTP (ESMTP) notifications can be configured to automatically deliver notifications to the default sender or nominated FROM address.

   - Check **Email notifications to sender** to enable mail server notifications.
   - In the **Notification types** section, select each of the events you want to be notified of by checking the relevant check boxes — **Success**, **Failure** and/or **Delay**.
   - In the **Notification contents** section, choose the notification to contain the **Complete email**, or **Headers only**.

6. To enable mail sent through the SMTP mail server to specify the sender’s address (to be used for replies and so on), Create!email must be notified of the FROM address to use. This will usually be done by adding the -FROM parameter to the header of the spool file (see Section 3.3.2).

   If this parameter is omitted from the spool file, Create!email will use the e-mail address entered in the **Default sender name** field. In order for the recipient’s Reply functions to be effective and the mail server to allow sending, the **Default sender name** must be entered as a valid e-mail address for a user of the SMTP mail server.

   **NOTE:** If the Default Sender is not specified and the spool file does not contain the FROM parameter, the SMTP mail server will be unable to deliver the e-mail. Create!email will record the error in the Create!email server error log, but is unable to deliver an e-mail error notification without a Default sender name.

7. Click **OK** to save your configuration settings and return to the Email Settings dialog. Click **OK** again to return to the main application window.

   **NOTE:** A successful delivery occurs when Create!email is able to deliver the message to your mail server. However, errors encountered by the SMTP mail server are not logged in Create!email.
2 Configure E-mail Account Information

Configure MAPI log-on:

1. After choosing MAPI as the Email transport, click the Configure button to display the MAPI Configuration options.

2. Enter the profile name used to access your MAPI e-mail account in the Profile field. This will be the same name you use to logon to your local e-mail client. To use the default MAPI profile you can leave this field blank.

3. Enter the password required to access your e-mail account in the Password field. Leave this field empty if your e-mail application does not require a password.

4. Click OK to save your configuration settings and return to the Email Settings dialog. Click OK again to return to the main application window.

---

**NOTE**

MAPI is not a preferred protocol for production printing environments. If Microsoft Outlook or Outlook Express must be used then refer to Appendix B for specific configuration settings.

---

Configure VIM log-on:

1. After choosing VIM as the Email transport, click the Configure button to display the VIM Configuration options.

2. Enter the profile name used to access your VIM e-mail account in the Profile field. Note that the profile name is case sensitive and is not necessarily the same as the account name used by your VIM e-mail client.

---

**NOTE**

To confirm the actual profile name used by the Lotus Notes client application, log on to your Lotus Notes client and choose File>Tools>User ID from the Main menu. The user name displayed in the list of Names should be entered as the profile name in Create!email, using the exact case (e.g. Dan Thompson/NW/TRN).

---

3. Enter the password required to access your e-mail account in the Password field. Leave this field empty if your e-mail application does not require a password.

4. Click OK to save your configuration settings and return to the Email Settings dialog. Click OK again to return to the main application window.
2.5.4 Configure Windows Identity

The CfService continues to run on your e-forms server, even when you are logged off from the machine. However, when using the MAPI or VIM protocols to send mail through a local e-mail client, Create!email will need to impersonate a valid user who has access to the machine, and can connect to the required e-mail account. To do so, you will be required to configure the Windows identity.

If Create!email has been configured to use an SMTP mail protocol, it is not necessary to specify a Windows identity.

To specify a Windows identity on the e-form server:

1. Select Administration\Windows identity from the Main menu. The Windows System Logon Details dialog will be displayed.
2. Enter the details for a valid user account on the machine:
   - In the Logon as user field, type the name you normally use to log on to the e-forms server and open your e-mail client.
   - If you normally logon to a domain, rather than a local account, enter the name of the domain in the Domain field.
   - Enter the password to the account in the Password field.
3. Click OK to save the settings for the Windows identity.

When you are logged off from the e-forms server, Create!email can continue to send e-mails through the client application, using the identity provided.
2 Create!email server Management

2.6 Create!email server Management

The Create!email server application provides an on-screen log of incoming, processing and paused jobs, as shown in the illustration below:

![Figure 2-5: The Create!email screen log](image)

2.6.1 Create!email server Processing

By default Create!email server runs continuously and jobs are processed as they are received, automatically clearing the job from the screen log once it has finished. When Create!email server has been paused, it is possible to manage individual Create!email jobs from the screen log.

To pause or resume Create!email server processing:

1. The title of the Create!email server application reflects the current state of the program ([Running] or [Paused]).

   If the CfService has been stopped, the dialog title will display [Server not Running] and no jobs will be displayed on the screen log.

2. To pause Create!email server processing, from the Main menu, select `Server > Pause`, or click the Pause button on the toolbar (pause icon).

3. To resume Create!email server processing, from the Main menu, select `Server > Resume`, or click the Resume button on the toolbar (play icon).
2.6.2 Working with Queued Jobs

Whenever the Create!email server application is paused (Section 2.6.1), any jobs received will be held in the Create!email queue and be displayed in the screen log, with ‘Waiting’ appearing in the Status column.

Jobs that are currently waiting may be paused, resumed or cancelled, directly from the screen log.

To pause, resume or cancel pending jobs:

1. From the Create!email server screen log, right click on the job to pause, resume or cancel.
   The following shortcut menu will be displayed:

   ![Figure 2-6: Manage waiting jobs]

2. From the shortcut menu, select a command to Pause this job, Resume this job or Cancel this job as appropriate:
   - You can also choose Resume all paused jobs to restart any paused jobs in the screen log. This does not resume Create!email server processing if Create!email server is currently paused (Section 2.6.1).
   - Select Cancel all jobs to prevent processing of any jobs currently in the Create!email server screen log and remove them from the queue.

3. If necessary, resume Create!email server processing by clicking the Resume button in the toolbar ( ).
2 E-mail a Test Merge Document

2.6.3 The CfService Generic Service

The CfService runs independently of Create!email server, accepting jobs from the Create!email queue and passing them to the relevant transport protocol for distribution. The service is automatically started when Create!email is first installed and will be automatically restarted whenever your system is started.

If the service stops running, Create!email server will be unable to send its jobs for processing and the service must be restarted in order to continue. Create!email server will display [Server not running] whenever the service has stopped. The service may be restarted from the Create!email server menu when required.

To start the CfService from Create!email server:

1. Select File/Start service from the Main menu.
   Once the service has been started, Create!email server will display [Running] in the title bar (even if Create!email server was previously paused, the queue will be resumed once the service is restarted).

2.7 E-mail a Test Merge Document

When you have finished configuring Create!email, you can test that it is working correctly by performing the built-in test. A sample message is sent to an e-mail address you nominate, using the e-mail protocol you defined earlier (Section 2.5).

2.7.1 Create!email Test

The Create!email test feature sends a simple message to a nominated e-mail address. The message contains the TO parameter which is automatically replaced by the e-mail address you type. It also contains a SUBJECT and a BODY. No attachments are included and no merging takes place before-hand.
To perform the Create!email test:

1. Select **File ▶ Test** from the Main menu.
   The CpPrintFile dialog will be displayed.

   ![CpPrintFile dialog](image)

   **Figure 2-7: Send a test e-mail using the CpEmailPort printer**

2. The test feature automatically selects the Create!email test file (emailtest.txt) and displays it in the **File** field. It is recommended that you do not change this file.

3. In the **Printer** section, select the printer queue that you configured for use with Create!email ([Section 2.4](#)). Only printer queues that are attached to the **CpEmailPort** output port can be selected from the list.

4. In the **To** field, enter the e-mail address to send the test e-mail to.

5. Click **Print** to complete the test.
2 E-mail a Test Merge Document

Assuming that your configuration is correct, a message similar to the following will appear in your e-mail account’s Sent folder (or the Outbox if the e-mail software is not connected):

![Image of e-mail message](image.png)

Figure 2-8: An example of the completed test e-mail

If the test e-mail did not complete as expected, Create!email will inform you of any errors using the method that you selected earlier (Section 2.5.2). If you wish to examine the Create!email job log for further information, refer to Section 2.8 for details.

2.7.2 Merged E-mail Test

To test that Create!form server is correctly merging and then e-mailing your documents, you must ensure that the Create!email queue is also defined as a Create!form server queue (Section 2.4.2). The merge test file contains parameters to specify an invoice form project for merging and to attach the merged file with the name New World Invoice Sample.pdf.

Create!form server must be able to locate the sample invoice form project which was installed with Create!form server. Refer to your Create!form server User Guide for instructions on specifying a form project directory.
To perform a merged Create!email test:

1. From the Main menu, select File→Test.  
   The CpPrintFile dialog will be displayed.

2. Click the Select button to choose a different file to print. Select the file called mergetest.txt if you are using Create!form server (Windows) or select mergetest.pdf if you are using Create!form server for JDE OneWorld. Click Open to select the file.

3. In the Printer section, select the printer queue that you configured for use with Create!email and Create!form server (Section 2.4). Only printer queues that are attached to the CpEmailPort output port can be selected from the list.

4. In the To field, enter the e-mail address to send the test e-mail to.

5. Click Print to complete the test.

   A sample invoice form will be merged and sent to the printer queue. As the queue is attached to the CpEmailPort output port, it will then be processed by Create!email and sent to your e-mail software.

---

If the test did not complete as expected, Create!email will inform you of any errors using the method that you selected earlier (Section 2.5.2). If you wish to examine the Create!email job log for further information, refer to Section 2.8 for details. You can also find error information in the Create!form server job log if required (Create!form server User Guide for more details).
2.8 Managing Create!email Logs

Create!email records different types of activity by maintaining a number of daily log files. Log files are stored in the Temp directory of the Create!form installation directory, this location can be modified by the user, see Section 2.8.1 for information on modifying log file settings.

- The Trace log records all Create!email events, except success.
- The Success log records all Create!email successful events.
- The Service log records all the activity of the service (CfService).

To view the Create!email logs:

1. From the Create!email Main menu, select View Trace log..., Success log..., or Service log....

2.8.1 Log File Settings

Using Create!email server, a user can specify global log file settings that apply to all Create!form International products, as well as choosing which events should be recorded specifically for Create!email server.

To make changes to the Create!email log file settings:

1. Select Administration Log file settings from the Create!email Main menu. The Log file settings dialog will be displayed.

   Any changes made to fields in the global log file settings section will come into effect for all other Create!form International products.

2. Click Browse to change the location for storing log files in the Path field.
3. Enter the number of days to store log files. Create!email stores log files for a specific period of time, depending on what is set in the Remove log files after days field. By default, this value is set to 7 (i.e., remove log files that are older than 7 days). Selecting 0 in this field will cause log files to be removed at the end of the day on which they were created. Previous log files have the date appended to the filename, for example: CFOWServeryyymmdd.log where yyyymmdd represents the log file date.

4. Select whether to log warning messages if the disk is nearing capacity. By default, Create!email server places a message in the log file when the disk becomes 90% full. If you do not want these messages logged, enter 0 in the Log a warning message if disk is full field.

5. Select whether to stop logging if the disk is nearing capacity. By default, Create!email server will no longer store log files once the disk is 95% full. To stop all logging, enter 0 in the Stop logging if disk is full field.

6. Set the level required for logging, in the Create!email events to be recorded section. Changes to these settings do not affect any other Create!form International products. To specify which events should be recorded within the log files, choose from the available options.

7. Click OK to finish defining Create!email log file settings.

2.9 Unprocessed Jobs

Any reports that are submitted to the e-forms server for merging when Create!email server is not running will remain in the $spool directory of the Create!form installation directory on the e-forms server.

When Create!email server is restarted, any jobs in the $spool directory will be resubmitted and processed automatically.
2 Failed Jobs

2.10 Failed Jobs

Spool file jobs will fail when Create!email cannot perform a requested action, such as attach a specified file or the e-mail message does not contain a TO parameter. Before attempting to re-send any failed jobs, you should first ensure that attachments, templates and other external files are identified correctly with the job, including the full path and filename where required.

From within Create!email server, you can:

- Retry any failed jobs by choosing Server ⇒ Reprocess failed jobs from the Main menu.
- Remove any failed jobs by choosing Server ⇒ Delete failed jobs from the Main menu.
Sending E-mails

This chapter provides details about specifying Create!email job parameters within a spool file document using a job header.

The following topics are covered:

- The Create!email Job Header
- E-mail Addressing
- E-mail Message Contents
- Originating File
- Other Attachments
- Templates and Placeholders
- Working with Templates
- Duplicate Header/Template Parameters
3 Overview of Job Headers

3.1 Overview of Job Headers

A job header is simply information that is attached to a spool file that sets out the parameters to use when processing a job. In order for Create!email to process a spool file, certain information is required, such as the name of the recipient. When spool files are to be merged prior to e-mailing, Create!form server parameters must also be included that identify such things as the form project name. The inclusion of job header parameters within a spool file does not affect the output of the printed document in any way.

A concise list of all available parameters that can be used in e-mail job headers can be found in Appendix A.

Figure 3-1: Process used by Create!form server and Create!email
The most efficient method of attaching parameters to spool files is to use the companion product, Create!stream. With Create!stream, you can define job details for text files and OneWorld PDF files and have parameters automatically sent with the job to Create!form server.

For information on defining Job Headers using Create!stream, please refer to the Create!stream User Guide.

3.2 The Create!email Job Header

If sending unmerged jobs to Create!email (i.e. the jobs do not need to be processed by Create!form server first), the spool file must indicate the start and end of the e-mail parameters by enclosing all e-mail parameters within the `%cpEmailBegin` and `%cpEmailEnd` statements.

The following shows the basic requirements of the minimal spool file:

```
%cpEmailBegin
%cpEmail:- Additional Create!email parameters here
%cpEmailEnd
[spool file data...]
```

Important: All spool file parameters are case-sensitive.

When Create!email receives the job for processing, it reads each of the additional `%cpEmail` parameters for processing instructions.

Alternatively, the e-mail parameters themselves may be embedded within Create!form server parameters, as shown in the following example:

```
%cpBegin
%cpEmail:- Additional Create!email parameters here
%cpParam:-sInvoice
%cpEnd
[spool file data...]
```
3 E-mail Addressing

In this case, Create!form server processes the job first, according to each of the %cpParam statements. The %cpParam statements are then removed and the job is forwarded onto Create!email for processing according to the %cpEmail instructions.

For additional information about the parameters available for Create!form server processing, please refer to your Create!form server User Guide.

3.3 E-mail Addressing

In order to create a complete e-mail message, the spool file must include a TO parameter that specifies the e-mail address of the recipient. A failed e-mail occurs when the TO parameter is not specified (Section 2.5.2).

The recipient e-mail address is specified using the following statement:

%cpEmail:-TOrecipient1@address.com

The e-mail address entered here will appear in the TO line of the e-mail message when it is created. Multiple recipients can be entered by including additional TO statements.

You can also provide additional recipients using the CC and BCC parameters:

%cpEmail:-TOrecipient1@address.com
%cpEmail:-TOrecipient2@address.com
%cpEmail:-CCrecipient3@address.com
%cpEmail:-CCrecipient4@address.com
%cpEmail:-BCCrecipient5@address.com
The above lines would produce an e-mail header similar to the following:

<table>
<thead>
<tr>
<th>To</th>
<th><a href="mailto:jakejake@address.com">jakejake@address.com</a>, <a href="mailto:victorvictor@address.com">victorvictor@address.com</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>CC</td>
<td><a href="mailto:jakejake@address.com">jakejake@address.com</a>, <a href="mailto:victorvictor@address.com">victorvictor@address.com</a></td>
</tr>
<tr>
<td>Bcc</td>
<td><a href="mailto:jakejake@address.com">jakejake@address.com</a></td>
</tr>
<tr>
<td>Subject</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3-2: Example of addressing an e-mail message

If duplicate e-mail addresses are encountered within a single job, Create!email will send only one e-mail to the address, no matter how many times it occurs.
3.3.1 Multiple Addressing on a Single Line

For each of the **TO**, **CC** and **BCC** parameters, you can also specify multiple e-mail addresses on the one line by identifying the *delimiter* that Create!email should recognize between addresses. The **DELIMITER** parameter must be entered in the job header prior to the first e-mail address, followed by the character or characters to be used as the delimiter(s). The following example shows how multiple addresses can be entered on a single line using a comma (,) as the delimiter:

```
%cpEmail:-DELIMITERS,
%cpEmail:-TOclient1@address.com,client2@address.com
%cpEmail:-CCAdministrator,Sales
```

3.3.2 Reply To Address

When Create!email is configured with the SMTP protocol, you can specify the reply address using the **FROM** parameter. If a valid e-mail address is not entered, the recipient will be unable to reply to the message using their e-mail software’s Reply function.

> For VIM and MAPI protocols, the Reply To address is always to the account name configured within Create!email.

The sender’s e-mail address can be specified using the **FROM** parameter:

```
%cpEmail:-TOrecipient@address.com
%cpEmail:-FROMsender@address.com
```

After receiving a message using the above parameters, the recipient can use the Reply function to send a message to `sender@address.com`.

> If the **FROM** parameter is not specified in the job header, Create!email will use the Default sender name entered in the SMTP Configuration dialog (Section 2.5.3).
3.4 E-mail Message Contents

To provide the contents of the e-mail message, the Subject (S), Body (BODY) and Signature (SIGNATURE) parameters are used as shown:

\[
\%\text{cpEmail}:-\text{TO} \text{recipient@address.com} \\
\%\text{cpEmail}:-\text{S} \text{Subject of e-mail} \\
\%\text{cpEmail}:-\text{BODY} \text{Text to appear in body} \\
\%\text{cpEmail}:-\text{SIGNATURE} \text{Finish with a signature}
\]

The text that follows the -S parameter will be placed in the Subject field of the outgoing e-mail message. The text following the -BODY parameter appears as the message body. The text for the -SIGNATURE parameter is added immediately after the body text in the e-mail, with one blank line automatically inserted between the two.

The above e-mail parameters would produce the following e-mail message:

---

**Figure 3-3:** An example of using subject, body and signature parameters in an e-mail

---
3 E-mail Message Contents

3.4.1 Multi-Line Body and Signature

Both the BODY and SIGNATURE parameters are capable of defining text consisting of multiple lines, including line breaks if required. The text for the parameter continues until the next occurrence of a %cEmail tag, the %cEmailEnd tag or the end of the file, whichever comes first.

The following shows an example of a multi-line body and signature specified within the header:

```
%cEmailBegin
%cEmail:-TOrecipient@address.com
%cEmail:-SSubject of e-mail
%cEmail:-BODYYBody text can appear over multiple lines of the header. The body parameter continues until the next parameter occurs.
%cEmail:-SIGNATURECompany Name
Address
Phone
Fax
%cEmailEnd
```

In the above example, the body text will be a single 3-line paragraph, while the signature will contain 4 lines, each separated by a line feed.

It is also possible to include placeholders within the BODY and SIGNATURE parameters, that are evaluated at the time of processing. For more information on using placeholder substitution, refer to Section 3.8.2.

Inserting blank lines in a multi-line parameter:

Create!email automatically strips out any blank lines that appear within the job header, including those that are created by a carriage return used in the BODY or SIGNATURE parameters. To ensure that blank lines used with these parameters are maintained after processing, simply include at least one character (such as a SPACE) on the blank line. Lines that contain at least one character will not be removed.
3.5 Originating File

Create!email automatically includes the original spool file as an attachment, and provides the ability to attach additional files to the e-mail message. By default, the originating file is attached to the e-mail message using the temporary filename assigned to it by Create!email.

Unfortunately, a filename such as 1106200015513500048.CPEmail may be rather meaningless to the e-mail recipient. To overcome this, a name can be assigned to the spool file using the **THIS** parameter as shown:

```
%cpEmail:-TOrecipient@address.com
%cpEmail:-SSubject of email
%cpEmail:-BODYText to appear in body
%cpEmail:-THISYour Invoice.pdf
```

---

**NOTE**

When PDF files are generated for use with Create!email, bar codes and MICR font will not be visible on-screen if the form project originates from Create!form for JDE OneWorld.

---

3.5.1 Omitting the Originating File

In most cases, the original OneWorld or Text spool file is firstly processed by Create!form server in order to produce a merged PDF document. This document is then sent as the originating file to Create!email. You can also eliminate the originating spool file from the e-mail message using the **NOSPOOL** parameter:

```
%cpEmail:-TOrecipient@address.com
%cpEmail:-SSubject of email
%cpEmail:-BODYText to appear in body
%cpEmail:-NOSPOOL
```

In the above example, the recipient message will contain a subject line and body text, but will not contain any attachments.
3 Originating File

3.5.2 Expanding the Original File

Rather than include the original spool file as an attachment, a spool file may be expanded within the body of the e-mail using the \(<\text{InsertSpool}\>) tag. The tag is placed within the \texttt{BODY} parameter at the location where you want the spool file to begin.

The following shows an example of using the \(<\text{InsertSpool}\>) tag to expand a text file:

\begin{verbatim}
%cpEmail:-TOrecipient@address.com
%cpEmail:-SSubject of email
%cpEmail:-BODYIn response to your query, \(<\text{InsertSpool}\>)
Thank-you for your enquiry.
\end{verbatim}

After processing, the e-mail message will appear similar to the following (depending upon the actual contents of the original spool file):

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{sample_email.png}
\caption{Sample e-mail after using the \(<\text{InsertSpool}\>) tag}
\end{figure}

In order for text to be expanded within the body of the e-mail message, Create!email checks the contents of the original spool file to determine if it:

- is a PDF file, a PostScript file, or a HTML file;
- contains additional control characters (other than Carriage Return, Line Feed, Form Feed and Tab) within the text.

If the original spool file meets any of the above conditions, it will be sent as an attachment regardless of the \(<\text{InsertSpool}\>) file tag.

---

\textbf{NOTE}

If the \texttt{COPYTO} parameter (\textit{Section 3.5.3}) occurs within the job header, the \(<\text{InsertSpool}\>) tag is replaced with a blank string.

In order for text to be expanded within the body of the e-mail message, Create!email checks the contents of the original spool file to determine if it:

- is a PDF file, a PostScript file, or a HTML file;
- contains additional control characters (other than Carriage Return, Line Feed, Form Feed and Tab) within the text.

If the original spool file meets any of the above conditions, it will be sent as an attachment regardless of the \(<\text{InsertSpool}\>) file tag.
3.5.3 WebLink Feature

Create!email’s weblink feature uses the **COPYTO** parameter to automatically store the original spool file in a defined directory and filename location, instead of attaching it to the e-mail message. In its place, the e-mail message can contain the URL address of the new spool file location.

The **COPYTO** parameter specifies the filename and location that the original spool file should be copied to. If a duplicate filename is found in the same location, Create!email will automatically generate a unique filename for the document by appending numbers to the filename. If this occurs, all occurrences of the specified filename within the job header or template will be replaced with the new unique filename.

**Weblink example:**

The following example uses the **COPYTO** parameter to copy the original spool file to the inet\invoices directory, with the name invoice000.pdf. The Body parameter (-**BODY**) identifies the URL to the same file.

```
%cpEmailBegin
%cpEmail:-COPYTOc:\inet\invoices\invoice000.pdf
%cpEmail:-BODYYour invoice can be found at http://web/invoices/invoice000.pdf
%cpEmailEnd
```

If c:\inet\invoices\invoice000.pdf doesn’t exist then it will be copied to that filename and the e-mail will contain the following text:

Your invoice can be found at http://web/invoices/invoice000.pdf

If the file already exists it will be copied to the same location but with a unique filename. The e-mail will then contain the following text:

Your invoice can be found at http://web/invoices/invoice000-000000.pdf

All occurrences of the weblink filename within the job header and the template will be replaced with the new filename.

---

*In all cases where the COPYTO parameter is used within a job header, the original spool file will not be attached to the e-mail message, nor inserted using the <<InsertSpool>> tag.*
Placeholder substitution:

It is also possible to use Create!email’s `PLACEHOLDER` parameter in conjunction with the `COPYTO` parameter to dynamically assign the new spool file location to different occurrences within the spool file header and/or template.

The placeholders parameter (P) defines a placeholder variable which is replaced with the nominated text at every occurrence of that name within the header and/or template. The placeholder parameter uses the following format:

```
%cpEmail:-Pplaceholder_name=replacement text
```

For example, the following line creates a placeholder with the name ‘Department’ and assigns it the replacement text of ‘Accounts’:

```
%cpEmail:-PDepartment=Accounts
```

In order to use the substitution text within the job header or template, the placeholder tag is entered at the location you want the text to occur. In the following example, the subject line will automatically substitute the word ‘Accounts’ where the `<<Department>>` placeholder variable occurs:

```
%cpEmail:-PDepartment=Accounts
%cpEmail:-S A message from <<Department>>
```

The resulting subject line would thus read: A message from Accounts.

To use placeholders with the `COPYTO` parameter, define a ‘Location’ placeholder that identifies the URL to the same file, and refer to it as required within the message using the `<<Location>>` tag. For example:

```
%cpEmailBegin
%cpEmail:-COPYTO c:\inet\invoices\invoice000.pdf
%cpEmail:-PLocation=http://web/invoices/invoice000.pdf
%cpEmail:-BODY Your invoice: <<Location>>
%cpEmailEnd
```

The body of the e-mail message will then contain the following text:

```
Your invoice: http://web/invoices/invoice000.pdf
```

More information on using placeholders within templates and job headers can be found in Section 3.9.4.
3.5.4 Sending a Secure File

Create!email is able to send the original spool file document as a secured PDF attachment by incorporating Acrobat Distiller security settings into the job header for processing by Create!form server.

In order to process Distiller security settings, you must be using Create!form server (Windows) version 2.3. This feature is currently not supported by Create!form server for JDE OneWorld.

Once the file has been processed by Create!form server and sent to the Create!email output port (CpEmailPort), the secured file will become the original spool file attachment for the e-mail message, according to the Create!email parameters in the job header.

**Defining Distiller security parameters:**

Create!form server recognizes the following header parameters for generating a secure PDF file:

- `%cpParam:-distupUser Password`
- `%cpParam:-distmpMaster Password`
- `%cpParam:-distsonnnn(Security Options)`
The following table describes the use of each parameter:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>distup</td>
<td>Specify the User Password required to open the document. For example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>%cpParam:-distupNewWorld</td>
<td></td>
</tr>
<tr>
<td></td>
<td>On attempting to open the PDF file attachment, the user will be prompted for the user password. The password is case-sensitive.</td>
<td></td>
</tr>
<tr>
<td>distmp</td>
<td>Specify the Master Password required for removing or modifying the security settings. For example:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>%cpParam:-distmpSysAdmin</td>
<td></td>
</tr>
<tr>
<td></td>
<td>If the user attempts to change the security password, save the document without security, or modify any security options, they will be prompted for the master password. The password is case-sensitive.</td>
<td></td>
</tr>
<tr>
<td>distso</td>
<td>Using a four-digit code, you can specify which features are enabled or disabled for the PDF file. The code is represented as follows:</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>Do not allow printing</td>
<td></td>
</tr>
<tr>
<td>0100</td>
<td>Do not allow changes to the document</td>
<td></td>
</tr>
<tr>
<td>0010</td>
<td>Do not allow adding or changing notes or form fields</td>
<td></td>
</tr>
<tr>
<td>0001</td>
<td>Do not allow selecting text and graphics</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Using a combination of the four codes, you can allow or disallow each of the features you require. The following example enables printing, but disables all other options:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>%cpParam:-distso0111</td>
<td></td>
</tr>
<tr>
<td></td>
<td>By combining security options with a master password, you can prevent users from being able to make changes to these settings.</td>
<td></td>
</tr>
</tbody>
</table>

If the COPYTO parameter is specified in the e-mail job header (Section 3.5.3), the copied file will also be secured using each of the security options that have been enabled. The original file is not sent as an attachment to the e-mail message.
3.6 Other Attachments

Using the AT parameter within a job file header, you can also include additional attachment files with your e-mail messages. The following shows an example of specifying an attachment:

%cpEmail:-TOrecipient@address.com
%cpEmail:-SSubject of email
%cpEmail:-ATCopyright.txt

To attach files from the default attachment directory:

In the above example, an attachment file is included using the AT parameter. The AT statement attaches the file called Copyright.txt from the default Create!email attachment directory:

\Create!form\Create!email\Attachment

---

The current attachment directory is displayed in the Create!email Properties dialog box, on the Directory Information tab.

---

To attach files from any directory:

You can also use an AT statement which specifies the full path to the attachment file. The full path must be used to attach files from any location other than the default Create!email attachment directory. For example:

%cpEmail:-ATC:\Notices\C1236.txt

This will attach the file called C1236.txt in the Notices directory on drive C.
3. Other Attachments

3.6.1 Attachment File Names

When an attachment is included with an e-mail message, the attachment continues to use its original file name by default.

To change the displayed file name of an attachment:

To change the name of the attachment when it is placed into the message, type the new name between << and >> symbols. For example:

\%cpEmail:-AT:\Notices\C1236.txt<<Legal.txt>>

The above example attaches the file called C1236.txt but renames it to Legal.txt in the e-mail message. The original file is not affected.

When using Create!stream, use the CONCAT function to create the appropriate entry in the job template definition file.

3.6.2 Multiple Attachments

Create!email supports only one attachment filename per AT parameter, however, you can attach multiple files to the e-mail message by incorporating multiple AT parameters in the job header or a template. For example:

\%cpEmail:-AT:\Notices\LEGAL.txt
\%cpEmail:-AT:\Notices\EULA.txt

The above example attaches the files called LEGAL.txt and EULA.txt from the NOTICES directory.

Attachment filenames should not be enclosed within double quotes, parentheses, or any other characters.
3.7 Templates and Placeholders

In addition to defining parameters within a spool file header, Create!email allows the use of Templates. A template lets the user define standard information once in a template file, rather than having to define the same information for every e-mail job. In addition, Create!email lets you include other templates within a template, allowing you to create generic text and information to be included in all jobs, whilst still retaining templates for specific purposes.

3.7.1 Template Parameter

Once created, a template can be used for any e-mail job by entering the TMPL parameter in the spool file header. For example:

```plaintext
%cpEmail:-TMPLAccount
```

By default, Create!email searches for templates with the extension .cpEmailTemplate stored in the default template directory–\<Createform install>\Create!email\Template. The full path and filename to the template file must be included to call files from any other location, or files that use a different extension. For example:

```plaintext
%cpEmail:-TMPLC:\Cust\July\Account.tpl
```

A template document can incorporate all of the available Create!email parameters. The following section describes how to design template files.
3 Working with Templates

3.8 Working with Templates

The following shows an example of a template document containing e-mail parameters:

```text
[TO]
  recipient1@address.com

[CC]
  recipient2@address.com
  recipient3@address.com

[BCC]
  recipient4@address.com

[FROM]
  sender@address.com

[SUBJECT]
  E-mail Subject

[BODY]
  Dear Customer,
  Please find attached your invoice for maintenance calls this month. Please read NOTICE.TXT for additional information.
[END_BODY]

[SIGNATURE]
  New World Company Pty. Ltd.
  123 Globe Street
  Globe City
[END_SIGNATURE]

[ATTACHMENTS]
  Legal.txt
  c:\Something.txt<<notice.txt>>
```

When Create!email reads the job file header, it locates the defined template file in the Template directory (`Create!form\Create!email\Template`), unless the full path name to the template is specified in the header (☞ Section 3.7.1).
3.8.1 Template Example

Information in the template provides the default parameters for the e-mail job. The following example shows a Create!email job header and the template that it calls:

C1238.TXT:
%cpEmailBegin
%cpEmail:-TOrecipient1@address.com
%cpEmail:-THISAccount.PDF
%cpEmail:-TMPLAccount.tpl
%cpEmailEnd
[spool file data]

Account.TPL:
[TO]
recipient2@address.com

[CC]
recipient3@address.com

[SUBJECT]
Monthly Account

[BODY]
Dear Customer,
Please find attached your maintenance account for the month of July. Please read the LEGAL notice for important information.
[END_BODY]

[SIGNATURE]
Regards,

New World Accounts
[END_SIGNATURE]

[ATTACHMENTS]
c:\Notices\C1235.txt<<LEGAL.TXT>>
3 Working with Templates

The resultant e-mail message would appear similar to the following:

![Email Message](image)

**Figure 3-5: A message resulting from the previous template**

### 3.8.2 Placeholder Substitution

Rather than provide all information as static text within a template, **placeholders** allow variable information to be inserted at runtime. Placeholders can be defined in the template in a [PLACEHOLDERS] section:

**Account.TPL:**

```plaintext
[BODY]
Dear <<Customer>>
Attached is your invoice for the month of <<Month>>
[END_BODY]

[PLACEHOLDERS]
Customer=Shady Spot
Month=July
```

When the template is executed, `<<Customer>>` is replaced with ‘Shady Spot’, and `<<Month>>` is replaced with ‘July’. These values will be used as the default placeholder values, unless the `P` parameter has been used to define the same placeholders in the job header as described below.
Placeholders used in the job header:

Placeholders defined within a template are considered to be default values unless defined in the header. Therefore, any placeholder values in the job header will override the default values defined in the template file. Placeholders can be defined within the job header using the `P` parameter to assign the variable information for the template. For example:

```
C1238.TXT:
%cpEmailBegin
%cpEmail:-TOrecipient1@address.com
%cpEmail:-TMPLAccount
%cpEmail:-PCustomer=Shady Spot
%cpEmail:-PMonth=July
%cpEmailEnd
[spool file data]
```

Placeholders allow you to have text within the body of the template determined at runtime by extracting information from the originating spool file. In the above example, the `Customer` and `Month` fields in the `Account.cpEmailTemplate` file will be filled when the spool file is sent.

*NOTE*

*Placeholder substitutions can also be referred to within the BODY, SIGNATURE and S parameters in the job header. For example, to use the Month placeholder value in the Subject line:*

```
%cpEmail:-SYour account for <<Month>>
```
3.8.3 Embedded Templates

In addition to specifying a template to use from the job header (using the TMPL parameter), a template may be called from within a template file using the [INCLUDE] section. In the following example, the header specifies the ‘Accounts’ template, which in turn includes the ‘Company’ template:

C1238.TXT:
%cpEmailBegin
%cpEmail:-TOrecipient1@address.com
%cpEmail:-TMPLAccounts.tpl
%cpEmail:-PMonth=July
%cpEmailEnd

[spool file data]

Accounts.TPL:
[CC]
accounts@address.com

[SUBJECT]
Monthly Account

[BODY]
Dear Customer,
Please find attached your maintenance account for the month of <<Month>>. Please read the LEGAL notice for important information.
[END BODY]

[INCLUDE]
Company.tpl

[ATTACHMENTS]
c:\Notices\C1235.txt<<LEGAL.TXT>>

Company.TPL:
[CC]
sysadmin@address.com

[SIGNATURE]
New World Company
15 Globe Rocks Road
GlobeWorld
[END_SIGNATURE]
In the previous example, the ‘Company’ template specifies an additional CC address and the Signature block. By separating company-wide or global information into a secondary template, any changes that may be required to a company address or other detail need only be made once in a single template file—there will be no need to modify all your Create!email templates.

**Evaluating template parameters:**

Template parameters are always evaluated in the order of their occurrence. An included template file is evaluated immediately it occurs within a template. When finished evaluating the included template, Create!email continues processing the calling template with the line immediately following the included file.

If a duplicate parameter exists in multiple templates, the parameter last read is the one used by Create!email. The next section contains further details on whether duplicate parameters are replaced or appended.

### 3.9 Duplicate Header/Template Parameters

The same parameters can be assigned in both the job header, a template file and included template files, which can result in conflicting or duplicate entries. When this occurs, the following rules identify the priority for each type of parameter.

#### 3.9.1 Addressing Fields

The **TO** parameter must be defined in the job header, otherwise an incomplete message will be generated (see Section 2.5.2). However, the **TO** parameter may also be defined within the template file.

Where **TO**, **CC** or **BCC** are found in both the template and the spool file header, Create!email merges each of the parameters, thus ensuring the e-mail is sent to all nominated recipients. Thus, if a **CC** is specified in both the header and the template, the message will be CC’d to both addresses. If duplicate e-mail addresses occur, Create!email will send only one e-mail to the address, no matter how many times it occurs within the header and/or templates.
3.9.2 Message Content

When the S or BODY parameters appear in both the template and the job header, those in the header take priority and the ones defined within the template are ignored. If multiple BODY parameters are encountered within included templates, the last one encountered is used unless it is overwritten by that defined within a header.

The SIGNATURE parameter used in the spool file header always overrides the signature in a template. If multiple SIGNATURE parameters appear within included templates, the last one read is used unless the Signature parameter occurs in the header.

3.9.3 Attachments

If attachment files are defined in both the job header and the template files, all specified attachments will be included in the e-mail message.

3.9.4 Placeholders

Placeholders defined within the template are considered to be default values unless defined in the header. Therefore, any placeholder values in the job header will override the default values defined in the template file.

Any new placeholders that are defined within the job header or included templates are added, but if a placeholder with the same name occurs in both the job header and a template, the one in the job header takes precedence.
Appendix A

This Appendix provides a listing of all Create!email parameters and template file sections, with examples and detailed descriptions.
## A.1 Email Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>%cpEmail:-DELIMITERS</code></td>
<td><code>%cpEmail:-DELIMITERS</code></td>
<td>Specifies a list of one or more single character delimiters (e.g., a comma <code>,</code>) to be used when parsing email recipient addresses. Allows for multiple addresses on a single line, by specifying where they will be split.</td>
</tr>
<tr>
<td><code>%cpEmail:-TO</code></td>
<td><code>%cpEmail:-TO person1@here OR %cpEmail:-TO person2@there, person3@here OR %cpEmail:-#!EmailAddress#!</code></td>
<td>List of one or more direct email recipients. There may be more than one <code>%cpEmail:-TO</code> line specified in the header and each line may specify more than one email address. See <code>%cpEmail:-DELIMITERS</code> to specify the delimiters to be used when combining multiple addresses on one line. In the last example the name is being mapped from data in the spool file using Create!stream - specifically, the email address from the spool file. The <code>#!…#!</code> signifies something to be mapped to that value using Create!stream. This parameter is compulsory, if not used the email will fail.</td>
</tr>
<tr>
<td><code>%cpEmail:-CC</code></td>
<td><code>%cpEmail:-CC person1@here OR %cpEmail:-CC person2@there, person3@here</code></td>
<td>List of one or more carbon copy email recipients. There may be more than one <code>%cpEmail:-CC</code> line specified in the header and each line may specify more than one email address. See <code>%cpEmail:-DELIMITERS</code> to specify the delimiters to be used when combining multiple addresses on one line.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Example</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>----------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>%cpEmail:-BCC</td>
<td>%cpEmail:-BCCperson1@here OR %cpEmail:-BCCperson2@there,person3@here</td>
<td>List of one or more blind carbon copy email recipients. There may be more than one %cpEmail:-BCC line specified in the header and each line may specify more than one email address. See %cpEmail:-DELIMITERS to specify the delimiters to be used when combining multiple addresses on one line.</td>
</tr>
<tr>
<td>%cpEmail:-FROM</td>
<td>%cpEmail:-FROMperson1@here</td>
<td>SMTP Email only - provides a FROM email address. If not present Create!email checks for a default FROM address specified via the admin client.</td>
</tr>
<tr>
<td>%cpEmail:-S</td>
<td>%cpEmail:-Sthis is the subject</td>
<td>Subject line for this email. The subject may contain placeholders using the &lt;&lt;placeholder&gt;&gt; syntax.</td>
</tr>
</tbody>
</table>
## Email Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
</table>
| `%cpEmail:-BODY` | `%cpEmail:-BODY This is the body text 
This is line2 of Body Text 
This is line3 of Body Text 
OR 
%cpEmail:-BODY Dear <<nameplacehldr>> 
Here is your invoice <<invnoplacehldr>> 
Please pay with 7 days 
Regards, 
The Accounts Department. | Body text for this email. The body text may contain placeholders using the `<<placeholder>>` syntax. Body text may be multi-line: all following lines will be included in the body text until the next occurrence of `%cpEmail` or the end of the file, whichever occurs first. In addition, a special tag `<<InsertSpool>>` may be used to insert the spool file into the body of the email rather than having it sent as an attachment. There are two considerations for this:
- If the COPYTO parameter is specified (see below) the `<<InsertSpool>>` parameter is replaced with a blank string.
- The spool file is checked to determine if it:
  · is a PDF file
  · is a Postscript file
  · is a HTML file
  · contains control characters other than Carriage Return, Line Feed, Form Feed or Tab.
If the spool file falls into any of these four categories it will be sent as an attachment regardless of whether the body text contains the `<<InsertSpool>>` tag. In all cases the email that is sent will not contain the `<<InsertSpool>>` tag. |
| `%cpEmail:-SIGNATURE` | `%cpEmail:-SIGNATURE Mr. Smith 
Accounts Dept Manager 
Phone: 3333 4444 
Email: E.Smith@ABC.com` | Specifies text that will be appended to the body text as a signature. One blank line is inserted between the body text and the signature. The signature text may contain placeholders using the `<<placeholder>>` syntax. The same rules apply to the signature as apply to the body text (see above). |
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%cpEmail:-NOSPOOL</td>
<td>%cpEmail:-NOSPOOL</td>
<td>Specifies that there is no attachment on the spoolfile. Email parsing finishes when the %cpEmailEnd tag is found and the rest of the file is ignored.</td>
</tr>
<tr>
<td>%cpEmail:-AT</td>
<td>%cpEmail:-ATdoc123.pdf&lt;&lt;legalinfo.pdf&gt;&gt;</td>
<td>A list of one or more file attachments. The files must be accessible on the machine running Create!email. Format is name&lt;&lt;displayname&gt;&gt;. The filename may be a fully qualified path or not. If not the file is assumed to reside under the Create!form\Create!email\attachments folder.</td>
</tr>
<tr>
<td>%cpEmail:-THIS</td>
<td>%cpEmail:-THIS&lt;&lt;invnoplaceholder&gt;&gt;</td>
<td>If the spoolfile contains an attachment, this specifies the filename to be used in the email for the file.</td>
</tr>
<tr>
<td>%cpEmail:-P</td>
<td>%cpEmail:-Pinvnoplaceholder=#!invno!#</td>
<td>A list of one or more placeholders. Format is name=replacement text. In this example the placeholder would be mapped to data from the spool file through Create!stream - specifically the invoice number in the spool file.</td>
</tr>
<tr>
<td>%cpEmail:-COPYTO</td>
<td>%cpEmail:-COPYTOC:\storedemails&lt;&lt;filename&gt;&gt;</td>
<td>This specifies that the attachment within the spool file (all bytes following the terminating %cpEmailEnd tag) should be copied to the filename specified. If the filename already exists Create!email will search for a non-conflicting filename (it never overwrites an existing file). The non-conflicting filename is created by appending a -000000 and testing for existence. If a file with that name exists then -000001 will be appended, and so forth until a non-conflicting filename is found.</td>
</tr>
</tbody>
</table>
## Email Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%cpEmail:-TMPL</td>
<td>%cpEmail:-TMPL</td>
<td>The name of a template file to include when building the email job. Either a fully qualified pathname or the name of a file assumed to exist in <a href="create/form/Create/email/Template">Create!form\Create\email\Templates</a> or a subfolder of this folder.</td>
</tr>
</tbody>
</table>

### Template File Sections

<table>
<thead>
<tr>
<th>Section</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[TO]</td>
<td><a href="mailto:person1@here.com">person1@here.com</a></td>
<td>Sets a list of one or more extra TO recipients who will be included in each email using this template. These are cumulative with recipients specified in the spool file.</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:person2@here.com">person2@here.com</a></td>
<td></td>
</tr>
<tr>
<td>[CC]</td>
<td><a href="mailto:person3@here.com">person3@here.com</a></td>
<td>Sets a list of one or more extra CC recipients who will be included in each email using this template. These are cumulative with recipients specified in the spool file.</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:person4@here.com">person4@here.com</a></td>
<td></td>
</tr>
<tr>
<td>[BCC]</td>
<td><a href="mailto:person3@here.com">person3@here.com</a></td>
<td>Sets a list of one or more extra BCC recipients who will be included in each email using this template. These are cumulative with recipients specified in the spool file.</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:person4@here.com">person4@here.com</a></td>
<td></td>
</tr>
<tr>
<td>[FROM]</td>
<td><a href="mailto:sender@there.com">sender@there.com</a></td>
<td>Specifies the FROM address but is only used for SMTP email - other email transports will substitute the email address attached to the email profile in use.</td>
</tr>
<tr>
<td>[SUBJECT]</td>
<td>This is the subject line</td>
<td>Sets a default subject line for the email. Always overridden if the subject is specified in the spool file. The subject text may contain placeholders using the <code>&lt;&lt;placeholder&gt;&gt;</code> syntax.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Example</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>[BODY]</td>
<td>[BODY]Dear &lt;&lt;nameplaceholder&gt;&gt;, This is line 1 of the Body text</td>
<td>Sets a default body for the email. Always overridden if the body is specified in the spool file. All text including line breaks appearing between a [BODY] section marker and the next [END_BODY] section marker are copied as the body text. If no [END-BODY] is present all text to the end of the template file will be included in the body. The body text may contain placeholders using the &lt;&lt;placeholder&gt;&gt; syntax.</td>
</tr>
<tr>
<td></td>
<td>This is line 2 of the Body text</td>
<td></td>
</tr>
<tr>
<td></td>
<td>This is line 3 of the Body text</td>
<td></td>
</tr>
<tr>
<td>[ATTACHMENTS]</td>
<td>[ATTACHMENTS]standard attachmtdoc.pdf C:\location\specialattachmtdoc.pdf</td>
<td>A list of one or more files to be attached to an email using this template file. These are cumulative with attachments specified in the spool file. As with attachments listed in the spool file these may be fully qualified pathnames or partial paths which are then assumed to exist under Create!form\Create!email\attachments.</td>
</tr>
<tr>
<td>[PLACEHOLDERS]</td>
<td>[PLACEHOLDERS]nameplaceholder=Mr.Smith name2placeholder=#!ToName#!</td>
<td>A list of one or more placeholders. These are cumulative with placeholders specified in the spool file, except if placeholders are specified with the same name in both places (the one in the spool file is the one which will be used). The format is placeholdername=placeholervalue</td>
</tr>
<tr>
<td>[INCLUDE]</td>
<td>[INCLUDE]deptdetails.tpl C:\location\specialtemplate.tpl</td>
<td>Allows a template file to include a second template file. The included file is evaluated immediately and when evaluation is finished processing continues with text following the included file.</td>
</tr>
</tbody>
</table>
## Email Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[SIGNATURE]</td>
<td>[SIGNATURE] Mr.Smith Accounts Manager Ph: 3333 4444 Email: <a href="mailto:Mr.Smith@ABC.com">Mr.Smith@ABC.com</a></td>
<td>Specifies text that will be appended to the body text as a signature. One blank line is inserted between the body text and the signature. If a signature is also specified in the spool file it overrides the one in the template file. All text between the [SIGNATURE] section and the [END_SIGNATURE] section including line breaks are copied verbatim. If there is no [END_SIGNATURE] section the signature will include all text up to the end of the file. The signature may contain placeholders using the &lt;&lt;placeholder&gt;&gt; syntax. If multiple signatures are encountered the last one becomes the default signature. The signature text specified in the template file is always overridden by the signature text specified in the header (if any).</td>
</tr>
<tr>
<td>[END_SIGNATURE]</td>
<td></td>
<td>Marks the end of a signature block.</td>
</tr>
</tbody>
</table>
Appendix B

This Appendix provides information on how to use the MAPI protocol to send email with Create!email.
B.1 MAPI Protocol and Create!email

MAPI was one of the first e-mail protocols to be developed. It is this first version of MAPI that is supported by Create!email (referred to as "Simple MAPI"). Several Microsoft products support this protocol (Outlook Express and Outlook, for example).

Outlook Express supports only the Simple MAPI protocol and so it is on this "lowest common denominator" that the support for MAPI in Create!email is built.

Simple MAPI communication has not been developed for many years and as such, only has a limited range of functionality. Due to its age, it is strongly recommended that the MAPI transport protocol provided by Create!email is not used for production e-mailing.

B.1.1 Create!email Transport Protocol

Applications such as Outlook and Outlook Express do not send e-mails themselves. Instead, they use more conventional e-mail protocols such as SMTP and VIM. It is therefore recommended that the SMTP or VIM transport protocol normally used by Outlook/Outlook Express is configured directly from within Create!email (i.e. configure Create!email to use a VIM or SMTP protocol, and do not configure Create!email with the MAPI protocol for sending mail through Outlook/Outlook Express). This will not only make e-mailing more reliable but also greatly increase the speed of processing.

B.2 Configure for Outlook Express

The following describes the recommended steps required to configure Create!email to send mail through Outlook Express.

B.2.1 Configure Create!email Settings

1. Start Create!email server and choose Administration\Configure email...
The Email Settings dialog will be displayed.
2. Select MAPI (Outlook/Outlook Express) in the Email transport section and click the Configure button.
The MAPI Configuration dialog will be displayed.
3. In the **MAPI Profile Name** field, type the name of the identity you normally use to logon to Outlook Express (see section B.2.2 for further information about identities). If required, enter the **Password** required for the identity.

4. It is necessary to ensure that a **Windows Identity** has not been entered in Create!email server. To verify that no Windows Identity exists, select **Administration**\>**Windows identity**.... Check that all fields within the Windows System Logon Details dialog are empty.

### B.2.2 Configure Outlook Express Identity

Create!email connects to Outlook Express using the identity you have entered in the MAPI configuration. This identity must exist in Outlook Express.

1. To verify the Outlook Express Identity name, start Outlook Express and choose **File**\>**Identities**\>**Manage Identities**. A list of available identities is displayed.

2. If required, you can create a new identity or use an existing identity with Create!email.

### B.2.3 Configure the Create!email Service

In order to correctly use Outlook Express to connect to your nominated mail server account, the Create!email service (CfService) must be configured to use a specific user logon. The user logon you enter must be a known network user for the system on which Outlook Express is installed. Generally, this will be the same network user name and password that you use to log on to your Windows system and send mail with Outlook Express.

To change the user account for the Create!email service:

1. From the Windows Control Panel, start the **Services** applet (this applet is found under **Administrative Tools** on Windows 2000).

2. In the list of services, select the service called **CfService** and choose **Action**\>**Stop** from the menu.

3. Double-click on **CfService** to display its **Properties**. Switch to the **Log On** tab and choose **This account** as the account to log on as.

   - In the **This account** field, click the **Browse** button to search for known user accounts on the network.
B. Configuration for Microsoft Outlook

- Once you have selected the correct user account, enter the password for the account in the **Password** field, and again in the **Confirm password** field.
- Click **OK** to finish.

4. Once completed, CfService must be restarted: choose **Action** ► **Restart** from the Services menu.

B.2.4 MAPI DLL’s

The following MAPI DLL’s were used during operation of Create!email with Outlook Express, and are known to work with the above configuration options. For more information, visit the Microsoft web site at:


<table>
<thead>
<tr>
<th>Name</th>
<th>Version</th>
<th>Directory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mapi32.dll</td>
<td>1.0.2536.0</td>
<td>WinNT\System32</td>
</tr>
<tr>
<td>Hmmapi.dll</td>
<td>1.0.2920.0</td>
<td>WinNT\System32\dllcache</td>
</tr>
<tr>
<td>Hmmapi.dll</td>
<td>1.0.2920.0</td>
<td>Program Files\Internet Explorer</td>
</tr>
</tbody>
</table>

B.3 Configuration for Microsoft Outlook

The following describes the recommended steps required to configure Create!email to send mail through Microsoft Outlook.

B.3.1 Configure Create!email Settings

1. Start Create!email server and choose **Administration** ► **Configure email**.... The Email Settings dialog will be displayed.

2. Select **MAPI (Outlook/Outlook Express)** in the **Email transport** section and click the **Configure** button. The MAPI Configuration dialog will be displayed.
3. In the MAPI Profile Name field, type the Outlook Profile name "MS Exchange Settings". Create!email will interact correctly with Microsoft Outlook as long as this profile is used.

4. Ensure the Password field is left empty, and click OK to continue.

**B.3.2 Configure Microsoft Outlook Identity**

Create!email connects to Microsoft Outlook using the MS Exchange Settings profile. This profile must exist in Outlook.

1. To verify the Microsoft Outlook profile exists, display the Windows Control Panel and double-click the Mail applet.

2. Click the Show Profiles button to display a list of current Mail profiles. Ensure that the MS Exchange Settings profile is displayed in the list. If the profile doesn’t exist, Add a new profile called MS Exchange Settings.

**B.3.3 Configure the Create!email Service**

In order to correctly use Microsoft Outlook to connect to your nominated mail server account, the Create!email service (CfService) must be configured to allow interaction with the desktop.

**To change the log on settings for the Create!email service:**

1. From the Windows Control Panel, start the Services applet (this applet is found under Administrative Tools on Windows 2000).

2. In the list of services, select the service called CfService and choose ActionuStop from the menu.

3. Double-click on CfService to display its Properties. Switch to the Log On tab and choose Local System account as the account to log on as.

4. Ensure the Allow service to interact with desktop check box is enabled (checked).

5. Once completed, CfService must be restarted: choose Action▶Restart from the Services menu.
### B.3.4 MAPI DLL’s

The following MAPI DLL’s were used during operation of Create!email with Microsoft Outlook, and are known to work with the above configuration options. For more information, visit the Microsoft web site at:


<table>
<thead>
<tr>
<th>Name</th>
<th>Version</th>
<th>Directory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mapi32.dll</td>
<td>1.0.2536.0</td>
<td>WinNT\System32</td>
</tr>
<tr>
<td>Hmmapi.dll</td>
<td>1.0.2920.0</td>
<td>WinNT\System32\dllcache</td>
</tr>
<tr>
<td>Hmmapi.dll</td>
<td>1.0.2920.0</td>
<td>Program Files\Internet Explorer</td>
</tr>
</tbody>
</table>
Index

%cpBegin, 3-3
%cpEmail, 3-3
%cpEmailBegin, 3-3
%cpEmailEnd, 3-3
%cpEnd, 3-3
%cpParam, 3-4
<<InsertSpool>>, 3-10

A

account
  log off e-mail account, 2-13
  MAPI logon, 2-16
  SMTP authenticated logon, 2-14
  VIM logon, 2-16
Acrobat Distiller
  installation, 2-2
  parameters, 3-13
  security settings, 3-13
Acrobat Reader
  installation, 2-7
  add new printer, 2-8
  address book, 1-7
  limitations, 2-11
  addressing e-mails, 3-4
AT
  default directory, 3-15
AT (attachment), 3-15
  multiple attachments, 3-24
attach file
  original spool file, 3-9
attachments
  AT parameter, 3-15
  cannot expand, 3-10
  default directory, 3-15
  directory, 3-15
  display name, 3-16
  file name, 3-16
  ignored for COPYTO, 3-11
insert original spool file, 3-10
  multiple files, 3-16
  multiple parameters, 3-24
  omit original spool, 3-9
  password protection, 1-5
authentication
  SMTP logon, 2-14
authorization code
  viewing, 2-5

B

BCC, 3-4
  multiple occurrences, 3-23
blind carbon copy, 3-4
BODY, 3-7
  blank lines, 3-8
  conflicting parameters, 3-24
  multiple lines, 3-8
  placeholder substitution, 3-21

C

cancel all jobs, 2-19
carbon copy, 3-4
CC, 3-4
  multiple occurrences, 3-23
CEmail.exe, 2-3
components
  Create!email, 1-7
  Create!form, 1-7
  Create!form server, 1-7
  e-mail client, 1-7
configure
  log files, 2-24
  MAPI transport, 2-16
  SMTP transport, 2-14
  transport protocol, 2-12
  VIM transport, 2-16
conflicting parameters, 3-23
conventions, 1-3
copy original spool file, 3-11
COPYTO, 3-11
cannot expand, 3-11
define URL, 3-11
filename, 3-11
location, 3-11
placeholder substitution, 3-12
CpEmailPort, 1-9
activate service, 2-7
attach to printer queue, 2-8, 2-9
create license key, 2-6
Create!email
cancel jobs, 2-19
cancel jobs, 2-19
client installation, 2-2
components, 1-5, 1-7
configure queue, 2-9
configure transport protocol, 2-12
Create!form server queue, 2-9
display license, 2-6
e-mail account, 1-4, 2-10
failed e-mails, 2-13
installation, 1-8
installation directory, 2-3
job headers, 1-10
license key, 2-6
log events, 2-25
log files, 2-14, 2-24
log off, 2-7
Lotus Notes support, 2-4
manual, 1-2
MAPI support, 2-4
merge test, 2-22
online help, 1-4
output port, 2-10
parameters, 1-10
pause jobs, 2-19
pre-requisites, 1-4
print spooler, 1-6
printer driver, 2-9
printer queue, 1-9
processing, 1-6
production, 1-11
requirements, 2-2
resume jobs, 2-19
screen log, 2-18
secure PDF, 1-5
sending documents, 1-2
setup, 2-3
SMTP support, 2-4
starting, 2-7
templates, 3-17
test job, 2-20
test results, 2-22
upgrading, 2-3
Create!email server
install, 2-2
pause queue, 2-10, 2-18
processing jobs, 2-18
restart queue, 2-10
resume, 2-18
Create!email service
auto-start, 2-7
server not running, 2-18
starting, 2-7, 2-20
Create!form
form design, 1-2, 1-5
transfer forms, 1-5
Create!form for JDE OneWorld
form design, 1-2, 1-5
Create!form server
%cpBegin, 3-3
%cpEnd, 3-3
%cpParam, 3-4
attach parameters, 1-10
merging, 1-2, 1-5
parameters, 3-3, 3-13
print processor, 2-8
printer queue, 1-9
properties, 2-8
secure PDF, 3-13
security settings, 3-13
sending documents, 1-3
sending documents, 1-3
test e-mail, 2-22
user guide, 1-2
Create!form transfer
transfer form resources, 1-2
transfer forms, 1-9
Create!send
  sending documents, 1-2
Create!stream
  attach parameters, 3-3
  installation, 1-8
  parameters, 2-3

---

D

definitions
  e-forms server, 1-2
delete failed jobs, 2-26
DELIMITER, 3-6
distmp (distiller master password), 3-13
distso (distiller security options), 3-13
distup (distiller user password), 3-13

---

E

e-forms server
  definition, 1-2
  domain name, 2-17
  logging off, 2-7, 2-17
  transfer resources, 1-9
  Windows identity, 2-17
e-mail account
  configure account, 2-14
  Lotus Notes, 2-16
  MAPI client, 2-16
  Microsoft Exchange, 2-16
  Microsoft Outlook, 2-16
  SMTP server, 2-14
  VIM client, 2-16
e-mail address, 1-7
  conflicting parameters, 3-23
  default sender name, 3-6
  duplicate addresses, 3-5
  multiple addresses, 3-6
  multiple parameters, 3-23
  multiple recipients, 3-4
  reply address, 3-6
  TO parameter, 2-13, 3-4
e-mail client
  account, 1-4
  local access, 2-11
  log off, 2-13
  Lotus Notes, 1-4
  Lotus Notes account, 2-16
  MAPI transport protocol, 1-4
  Microsoft Exchange, 1-4
  Microsoft Outlook, 1-4
  profile name, 2-16
  remote server, 2-2
  SMTP transport protocol, 1-4
  support for, 1-4
  transport protocols, 1-4
  VIM transport protocol, 1-4
e-mail message
  blank lines, 3-8
  contents, 3-7
  contents duplicated, 3-24
  incomplete, 2-13
  multiple lines, 3-8
  reply, 2-15
  sender, 2-15
  subject, 3-7
  test, 2-20
e-mail server
  accessing, 2-11
  direct access, 2-11
  SMTP protocol, 2-11
  transport protocol, 2-11
error notification
  contents, 2-13
  notify address, 2-13
errors
  e-mail test, 2-23
  failed jobs, 2-26
  notify, 2-13
  SMTP server, 2-15
ESMTP
  authenticated logon, 2-14
  notifications, 2-15
expand original file, 3-10

**F**

failed jobs, 2-13, 2-26
delete, 2-26
directory, 2-26
notify address, 2-13
reprocessing, 2-26
SENT folder, 2-13
to parameter missing, 3-4
form design
installation, 1-2
JDE OneWorld, 1-5
transfer form resources, 1-5, 1-9
transfer forms, 1-2
FROM, 3-6
limitations, 2-11

**H**

header
see job headers, 3-2
help
Acrobat Reader, 2-7
online help, 1-4
host system
sending documents, 1-2

**I**

install
Acrobat Reader, 2-7
Create!email, 1-8
Create!form server, 1-8
Create!stream, 1-8
installation
Administrator, 2-2
default directory, 2-3
e-mail client, 2-2
protocols installed, 2-4
requirements, 2-2

**J**

job headers
%cpBegin, 3-3
%cpEmailBegin, 3-3
%cpEmailEnd, 3-3
%cpEnd, 3-3
%cpParam, 3-4
<<InsertSpool>>, 3-10
AT parameter, 3-15
attaching, 2-3, 3-3
attachment file name, 3-16
BCC parameter, 3-4
BODY parameter, 3-7
CC parameter, 3-4
conflicting content, 3-24
conflicting parameters, 3-23
COPYTO, 3-11
DELIMITER parameter, 3-6
duplicate parameters, 3-23
e-mail parameters, 3-3
evaluating templates, 3-23
example template, 3-19
form project, 3-2
FROM parameter, 3-6
global templates, 3-23
identify parameters, 1-10
insert spool file, 3-10
multiple attachments, 3-16, 3-24
NOSPOOL parameter, 3-9
omit spool file, 3-9
overview, 1-10, 3-2
P (placeholder) parameter, 3-20
placeholder substitution, 3-21
S (subject) parameter, 3-7
SIGNATURE parameter, 3-7
spool file name, 3-9
template sections, 3-18
templates, 3-17
THIS parameter, 3-9
TMPL (template) parameter, 3-17
to parameter, 3-4
URL link, 3-11
weblink feature, 3-11

**L**

license
  display details, 2-6
license key
  authorization code, 2-5
  create, 2-6
  details, 2-6
  enter, 2-6
  permanent, 2-5, 2-6
  products enabled, 2-6
  requirements, 2-5
  temporary, 2-6
limitations
  address book lookup, 2-11
  e-mail client, 2-11
  FROM parameter, 2-11
  SENT folder, 2-11
log file
  directory location, 2-24
  disk space warnings, 2-25
  events to record, 2-25
  failed messages, 2-14
  global settings, 2-24
  remove old files, 2-25
  screen log, 2-18
  settings, 2-24
logon
  MAPI account, 2-16
  SMTP account, 2-14
  VIM account, 2-16
Lotus Notes
  installation, 2-2
  pre-requisites, 2-4
  profile name, 2-16
  support for, 1-4
  user ID, 2-16
  VIM transfer protocol, 1-4

**M**

mail protocol
  configure, 2-12
mail transfer protocol
  see transport protocol, 1-4
MAPI
  configure, 2-16
  default profile, 2-16
  FROM address, 2-11
  limitations, 2-11
  log on, 2-16
  profile name, 2-16
  reply to address, 3-6
master password
  secure PDF files, 3-14
merge
  form project directory, 2-22
  production printing, 1-2
  test e-mail, 2-22
Microsoft Outlook
  installation, 2-2
  MAPI transport, 1-4
Microsoft Exchange
  MAPI transport, 1-4
  multiple addresses, 3-6

**N**

NOSPOOL, 3-9
notifications
  Create!email errors, 2-14
  error e-mail, 2-13
  ESMTP notification types, 2-15

**O**

operating system, 2-2
P

P (placeholder), 3-12, 3-20
  conflicts, 3-24
  used in header, 3-21
parameters
  %cpEmail, 3-3
  %cpEmailBegin, 3-3
  %cpEmailEnd, 3-3
  %cpParam, 3-4
  <<InsertSpool>>, 3-10
  AT, 3-15
  attach template, 3-19
  attaching, 1-10
  BCC, 3-4
  blank lines, 3-8
  BODY, 3-7
  CC, 3-4
  conflicting, 3-23
  COPYTO, 3-11
  Create!form server, 3-2, 3-4
  Create!stream, 1-10
  DELIMITER, 3-6
  distmp (master password), 3-13
  distso (security options), 3-13
  distup (user password), 3-13
  duplicates, 3-23
  FROM, 3-6
  minimum, 3-3
  multiple attachments, 3-24
  multiple lines, 3-8
  NOSPOOL, 3-9
  overview, 1-10
  P (placeholder), 3-12, 3-20
  S (subject), 3-7
  SIGNATURE, 3-7
  THIS, 3-9
  TMPL (template), 3-17
  TO, 3-4
  pause Create!email server, 2-18
  placeholder substitution, 3-20
  placeholders
    conflicting, 3-24
    used with COPYTO, 3-12
  platform, 2-2
  port
    CpEmailPort, 2-9
    Create!email queue, 2-9
  pre-requisites
    e-mail client, 1-4
    printer operations, 1-4
  print processor, 2-8
  print spooler
    stopping, 2-3
  printer driver, 2-8
  printer queue
    add printer, 2-8
    attach CpEmailPort, 2-9
    configure, 1-9, 2-9
    create, 2-8
    Create!form server, 2-8
    driver, 2-8
    output port, 2-8, 2-9, 2-10
    print processor, 2-8
    processing for Create!email, 1-6
  printers
    number of licenses, 2-6
    profile name, 2-16

R

reply to address, 3-6
reprocess failed jobs, 2-26
resume Create!email server, 2-18

S

S (subject), 3-7
  conflicting parameters, 3-24
  placeholder substitution, 3-21
  security
    copied files, 3-14
    Create!form server requirements, 3-13
    disable changes, 3-14
    disable notes, 3-14
disable printing, 3-14
disable selection, 3-14
parameters, 3-13
PDF documents, 1-5
PDF passwords, 3-13
sender name
   SMTP, 2-15
sending documents
   Create!send, 1-2
service
   stopped, 2-18
Service log, 2-24
setup, 2-3
SIGNATURE, 3-7
   blank lines, 3-8
   multiple lines, 3-8
   placeholder substitution, 3-21
SMTP
   authenticated logon, 2-14
   configure account, 2-14
   default sender name, 2-15
   FROM address, 2-15
   FROM parameter, 3-6
   limitations, 2-11
   mail server errors, 2-15
   notifications, 2-15
   server name, 2-14
   test connection, 2-14
   transport protocol, 1-4
spool file
   Create!stream, 1-8
split files
   Create!email queue, 2-19
   expand original, 3-10
   job header, 3-2
   omit original, 3-9
   parameters, 1-10
   password protection, 3-13
starting Create!email, 2-7
subject line, 3-7
Success log, 2-24

T

templates
   [ATTACHMENTS], 3-18
   [BCC], 3-18
   [BODY], 3-18
   [CC], 3-18
   [END_BODY], 3-18
   [END_SIGNATURE], 3-18
   [FROM], 3-18
   [INCLUDE], 3-22
   [PLACEHOLDERS], 3-20
   [SIGNATURE], 3-18
   [SUBJECT], 3-18
   [TO], 3-18
   conflicting placeholders, 3-24
   design, 3-18
   directory, 3-17
   duplicate parameters, 3-23
   embedded, 3-22
   evaluating, 3-23
   example, 3-19
   extension, 3-17
   filename, 3-17
   global templates, 3-23
   include, 3-22
   overview, 3-17
   test Create!email, 2-20
   test message
   error log, 2-22
   merged test, 2-22
   perform test, 2-21
   results, 2-22
   THIS, 3-9
   TMPL (template), 3-17
   filename, 3-17
   TO, 3-4
   conflicting addresses, 3-23
   multiple occurrences, 3-23
   Trace log, 2-24
transfer forms, 1-2
transport protocol
  configure, 2-12, 2-16
  local e-mail client, 2-11
Lotus Notes, 1-4
MAPI, 1-4
Microsoft Exchange, 1-4
Microsoft Outlook, 1-4
SMTP, 1-4
SMTP e-mail server, 2-11
types of, 2-11
VIM, 1-4

unprocessed jobs, 2-25
upgrading, 2-3
URL placeholder, 3-12
user guide, 1-2
  conventions, 1-3
  online help, 1-4
user logon, 2-17
user name
  MAPI logon, 2-16
  SMTP logon, 2-14
  VIM logon, 2-16
user password
  secure PDF files, 3-14

view authorization code, 2-5
VIM
  configure, 2-16
  FROM address, 2-11
  limitations, 2-11
  log on, 2-16
  profile name, 2-16
  reply to address, 3-6

weblink, 3-11
weblink example, 3-11
weblink feature
  placeholder example, 3-12
Windows identity, 2-17