



**JetTrac Notify<sup>™</sup>  
Installation Guide  
User Guide**

**Version 1.1  
February 15, 2000**

## Version History

Version History		
January 25, 2000	1.0	Initial release of JetTrac Notify. Provides for FNF and E-mail output with different options for every job.
February 15, 2000	1.1	Enhanced how JetTrac Notify communicates with SMTP servers. Changed formatting options for subject line and body of email

### **JetTrac Notify™ Installation Guide and User Guide January 25, 2000**

Copyright© Pro Technology Automation, Inc., 2000  
All rights reserved.

#### **Trademarks**

JETFORM is a registered trademark and JetForm Central is a trademark of JETFORM Corporation.  
JetTrac is a trademark of Pro Technology Automation, Incorporated.

# TABLE OF CONTENTS

<b>1.0</b>	<b>INTRODUCTION</b> .....	<b>4</b>
1.1	PURPOSE .....	4
1.2	BACKGROUND.....	4
1.3	ORGANIZATION OF THIS DOCUMENT .....	4
1.4	TECHNICAL SUPPORT .....	4
1.5	TYPICAL JETFORM CENTRAL JOB FLOW USING JETTRAC NOTIFY.....	4
<b>2.0</b>	<b>INSTALLATION AND CONFIGURATION OF JETTRAC NOTIFY™</b> .....	<b>6</b>
2.1	INSTALLING JAVA .....	6
2.2	INSTALLING JETTRAC NOTIFY.....	7
2.3	CONFIGURING THE JETTRAC NOTIFY.INI FILE.....	8
2.4	CONFIGURING THE TASK IN THE JOB MANAGEMENT DATABASE .....	9
<b>3.0</b>	<b>RUNNING JETTRAC NOTIFY™</b> .....	<b>12</b>
3.1	SUBMITTING TRANSACTION FILES TO JETFORM CENTRAL.....	12
3.2	HOW JETTRAC NOTIFY™ IS TRIGGERED.....	12
3.3	WHAT JETTRAC NOTIFY™ DOES.....	12
3.4	JETFORM CENTRAL LOG FILE .....	12

## 1.0 INTRODUCTION

### 1.1 Purpose

This document is a guide to installing, configuring and using the JetTrac Notify™ Custom Agent. This document includes detailed instructions for installation and configuration in the JetForm Central Job Management Database.

### 1.2 Background

JetTrac Notify™ sends notification of JetForm Central events. JetTrac Notify can run a standard JetForm Central job task or can be triggered as a part of the JFError task to give you emergency notification if a job fails. Notification can come via printing or E-mail and is individually customizable for affirmative or error notifications for every separate job you run.

JetTrac Notify™ is written in Java so it will run on any platform that supports Java, including, among others, Windows, UNIX, and AS/400. See the *JetTrac Notify Installation and User Guide for AS/400* for help on AS/400 installation and configuration.

### 1.3 Organization of this Document

Section 1: Introduction

Section 2: Installation and Configuration Procedures for JetTrac Notify™

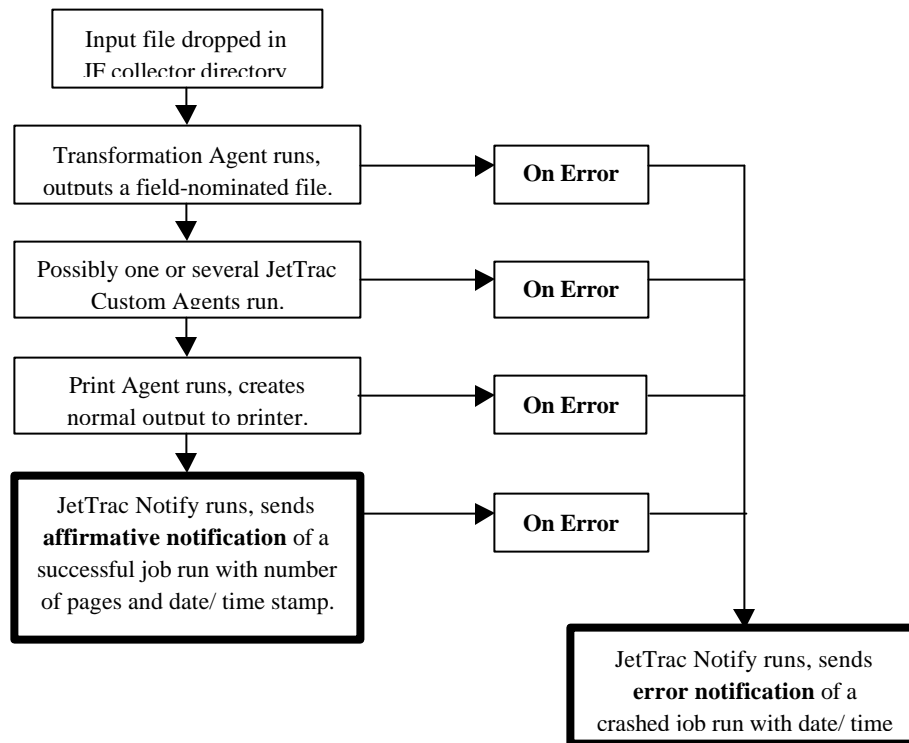
Section 3: Running JetTrac Notify™

### 1.4 Technical Support

If you need assistance in installing and configuring JetTrac Notify™, call Pro Technology Automation, Inc. at 805-527-1248 or email us at support@protechinc.com. Please note that the JetTrac Notify™ license fee does not cover configuration services and technical support so there may be an additional charge. Please ensure you read these instructions carefully before calling for technical support.

## 1.5 Typical JetForm Central job flow using JetTrac Notify

For most users, JetTrac Notify will either be the last task in a job (positive notification) or as the JFError task (error notification) or both. JetTrac Notify allows for distinct notification options in the event of an error or success. Each and every job can have different options as well. This example shows how JetTrac Notify might be implemented in a typical JetForm Central job.



## 2.0 INSTALLATION AND CONFIGURATION OF JETTRAC NOTIFY™

### 2.1 Installing Java

JetTrac Notify is a Java program. This requires that the Java Runtime Environment be installed before JetTrac Notify will run.

For Windows 95/98/NT, install JRE 1.2.2 (Newest as of this writing, any newer version will also work) for Windows. You can download this at:

<http://java.sun.com/products/jdk/1.2/jre/download-windows.html>

<http://www.javasoft.com/products/jdk/1.2/jre/download-windows.html>

<http://www.protechinc.com/storage/jre122win32.exe>

For AS/400 platforms, Java is an integral part of the OS/400, release 4.3 and above. If you are running an older version of OS/400, please contact your AS/400 support to find out how to run Java applications.

For Unix platforms, JRE may already be installed as part of the operating system. If it is not, contact the manufacturer of your server/workstation to find out where to download it.

For Sun Solaris 7, download JRE at:

<http://www.sun.com/solaris/jre>

For AIX, OS/2 or OS/400, download JRE at:

<http://www.ibm.com/java/jdk/>

For Digital Open VMS, Digital Unix 4.0x or Digital Windows NT Alpha, download JRE at:

<http://www.digital.com/java/download/>

For HP-UX, download 1.1.7 HP-UX 10.20 JRE at:

<http://www.hp.com/go/java>

## 2.2 Installing JetTrac Notify

After the JRE is installed, you are ready to proceed to the JetTrac Notify files themselves. You will receive an installation EXE file. Simply double click on the EXE file to unzip it and follow the instructions. The following files will be installed:

<u>Folder name</u>	<u>Filename</u>	<u>Description</u>
C:\jfsrvr\CustomAgents\JetTracNotify	RunJetTracNotify.bat	Batch file for testing
C:\jfsrvr\CustomAgents\JetTracNotify	JetTracNotify.jar	JetTrac Notify executable program file
C:\jfsrvr\CustomAgents\JetTracNotify	JetTracNotifyNom.ini	JetTrac Notify Nominal INI file
C:\jfsrvr\CustomAgents\JetTracNotify	JetTracNotifyError.ini	JetTrac Notify Error INI file
C:\jfsrvr\CustomAgents\JetTracNotify	notify.lic	JetTrac Notify license file for your computer
C:\jfsrvr\CustomAgents\JetTracNotify	JetTracNotifyUserGuide.pdf	– this document

## 2.3 Configuring the JetTracNotify ini files

There are two main steps to complete the configuration of JetTrac Notify. The first is to set up the JetTracNotify.ini files. The second is to set up the tasks and job steps in the Job Management Database in JetForm Control to call JetTrac Notify™ at the correct time.

There are two ini files for JetTrac Notify. Though used for different purposes, the files have the exact same structure. Each line inside the ini files refers to one job each. The lines are “pipe-delimited,” meaning that each different piece of data is separated by a pipe “|” character. One ini file contains the configuration for what to do when a job is completed successfully and the other contains the configuration for what to do when a job fails. The two files in this instance are JetTracNotifyNom.ini and JetTracNotifyError.ini, respectively, though you can name them whatever you want for your application.

Open JetTracNotifyNom.ini or JetTracNotifyError.ini file. A sample looks like this:

```
PackingSlip|E|JetForm has successfully printed %j% on %d% with %l% pages on the printer
%2%.
    |209.78.193.23|Dan@protechinc.com|dayl@protechinc.com,lori@protechinc.com|S||
APChecks|E|JetForm has successfully printed %j% on %d% with %l% pages on the printer
%2%.
    |209.78.193.23|Dan@protechinc.com|dayl@protechinc.com,lori@protechinc.com|S||
PickSlip|P|The %j% job has finished printing with %l%
pages. ||||packdone|c:\jfsrvr\data\pack.dat|
~~
Field 1:jobname
Field 2:P for print,E for E-mail,B for both
Field 3:the output string, with places left for the command-line variables, a j is the
job card,
    a d is the date/time, and numbers are the corresponding command-line parameters.
Field 4:SMTP Server (must be IP to avoid DNS lookup or ignored for FNF only output
Field 5:from: e-mail address or ignored for FNF only
Field 6:to: email address(es) (separated by commas) or ignored for FNF only
Field 7:string goes in the Subject line [S] or the body [B] of E-mail or ignored for FNF
only
Field 8:job card for a FNF output or ignored for E-mail only
```

Field 9:the filename for the FNF output or ignored for E-mail only

In the actual file, there would be only three lines above the double-tilde (~~). They are line-wrapped here out of necessity. Refer to the included ini files for further reference. The lines above the double-tilde (~~) are values that JetTrac Notify reads. The rest of the lines are internal documentation to prompt you for what goes on each line. The double-tilde (~~) serves to separate the code lines from the explanations. Pipes separate the sections of each line. If you are using print or E-mail notifications only, you do not need to enter data in the sections that do not pertain to the option you want. However, you must still leave the pipe characters so JetTrac Notify will know where to expect to find your options data. We will now explain what each section of the configuration lines does.

**Field 1** is the job name from the JetForm Central job management database. Because JetTrac Notify can be configured differently for each job, this field is necessary to identify which job is currently running or has just failed.

**Field 2** is a code to tell JetTrac Notify which options to run for this job. If you want to have JetForm Central only print a notification, enter a P here. If you want to have JetForm Central only E-mail a notification, enter an E here. If you want to do both options, enter a B here.

**Field 3** is the output string. This same string will appear in the E-mail, the printout or both. You can enter whatever you want here, with the addition of a few variables. If you want to have the job name in the output string, enter a %j% where you want it. If you want to have the date/time in the output string, enter a %d% where you want it. If want to have any JetForm Central variables appear in your output string, enter a %[number]% where you want it, where [number] is the number corresponding to the variable you entered on the command-line. For example, if you want to have the number of pages printed show up in your output string, enter -#1@NumberPages on the command line and enter %1% in the string here. JetForm Central will replace @NumberPages with the number of pages just printed and JetTrac Notify will place that number in your output string. See the instructions about command-line options in the next section for more information on using JetForm Central variables in the command line.

**Field 4** is the SMTP server address. This must be a numeric IP address number (ie 182.232.123.32) and not a pneumonic address (ie mail.intel.com). If you do not know your SMTP server address or its numeric IP address, ask you system administrator. In any case, verify that you have access and rights to an SMTP mail server before attempting to use JetTrac Notify, unless you plan to use it for print notifications only.

**Field 5** is the "From:" E-mail address. This will appear in the "From" line of E-mails received in relation to the job in **Line 1**.

**Field 6** is the "To:" E-mail address(es). This is the E-mail address or addresses to which you wish to send notifications in relation to the job in **Line 1**. If you wish to send the notification to more than one E-mail address, separate each address with a comma "," If you wish to send a notification to an alphanumeric pager, contact your pager's service

for information on how to send an E-mail to the pager. Usually it is in the form of [pagenumber]@paging-service.com (ie 8055342234@skytel.com).

**Field 7** is a code to tell JetTrac Notify to put the output string in the subject of the E-mail or the body. We put this in because some alphanumeric paging services require information to be either in the body or the subject of a message, or may possibly truncate a message if only in the subject. Enter an S for inserting the output string in the subject of the E-mail or a B for inserting the output string in the body of the E-mail message.

**Field 8** is for print output only. This is the job card that will be placed at the top of the field-nominated file output by JetTrac Notify. A job card is required to be at the top of the file for JetTrac Notify to be able to tell JetForm Central what job to run when print notification is desired.

**Field 9** is the name of the fully-qualified, field-nominated file to be output by JetTrac Notify. In most cases, this will include the path to the collector directory and any file name with the extension of .dat though may be different in your application.

## 2.4 JetForm Central Variables

JetForm has provided a number of built-in variables in JetForm Central that allow us to pass dynamic values. All JetForm Central variable names begin with the @ symbol. The prime example of a JetForm Central variable is @JobName. This variable is automatically assigned the value of the current job name being run by JetForm Central. Using this variable, we can pass to JetTrac Notify the name of the current job being run. This variable, @JobName, is the only variable required by JetTrac Notify. It is used to lookup the notification options for each individual job. You can use up to six other variables to pass to JetTrac Notify. While there are many choices listed in the manual, here is a list of the most common.

<u>Variable Name</u>	<u>Variable Description</u>
@JobName	The job name in Central (required for this application)
@JobID	A parameter passed with -aji in the initial job card
@DataFile	The input data file name
@PhysicalDev	The output device UNC
@NumberPages	The number of pages printed from central (will only be provided after a successful merge)
@DataFile	Fully qualified name of the transaction file
@LogFileName	The main JetForm Central Log File
@OtherJobTokens	Any "Token" information passed in the job card
@TaskID	Name of the Task from the Job Step
@UserID	Name of the user submitting the transaction, if available

## 2.5 Configuring the task in the Job Management Database

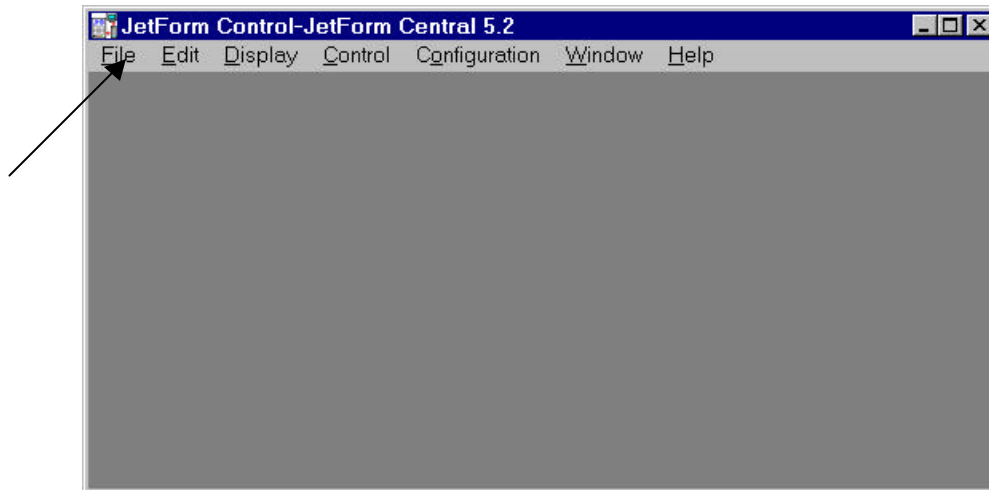
JetTrac Notify will take several command-line arguments:

```
<ini file> <log file> @JobName -#1<variable 1> -#2< variable 2> -#3< variable 3> -#4< variable 4> -#5< variable 5> -#6< variable 6>
```

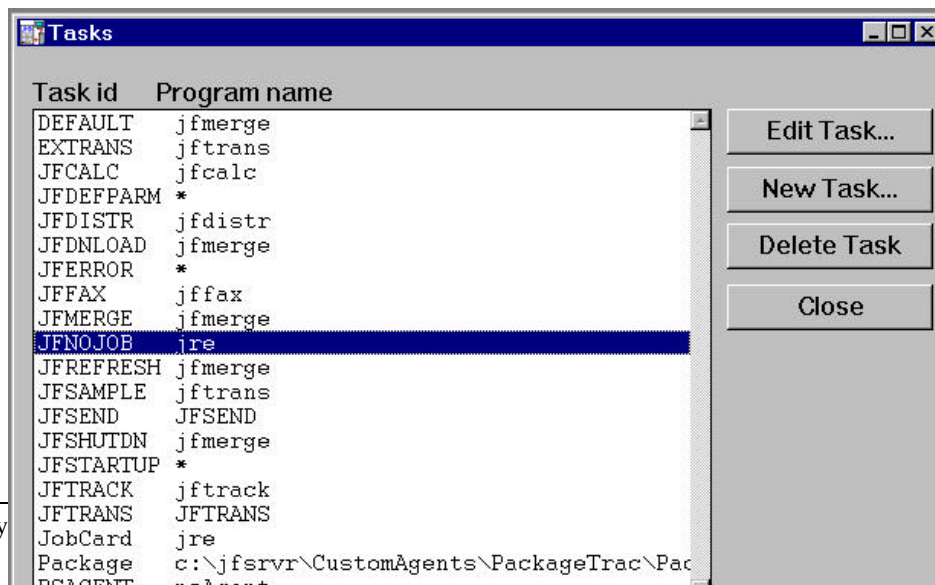
The INI file is the file with all the definitions as specified in Section 2.2. The log file is the fully qualified JetForm Central logfile name. The @JobName is a variable name that will cause JetForm Central to replace @JobName with the name of the current job. The rest of the command-line entries are optional. These extra entries provide the means by which other JetForm Central variables can be passed into your output string. For instance, if you want the number of pages printed to appear in your output string, you could enter -#1@NumberPages on the command line and then use %1% in your output string. Refer to the ini file section of this document for information on setting up the output string with JetForm Central variables. Each of the extra command-line entries must be preceded by a -#[number] in order for JetTrac Notify to recognize it. The variables must go in numeric order so that a -#3 must be preceded by a -#1 and a -#2 in that order. You can use up to six variable entries.

To add the JetTrac Notify custom agent in JetForm Control for Windows platforms:

1. Start JetForm Control by clicking Start, Programs, JetForm Central, JetForm Control (may be slightly different for your installation). You will get the following screen:



2. Add a new Task by clicking on File, Job Management Database, Tasks. You will get the following screen:



3. Create a new task by clicking New Task. Set up the Task Definition with the following information:

**Task ID:** JTNotify  
**Program name:** java -jar c:\jfsrvr\CustomAgents\JetTracNotify\JTNotify.jar  
**Program options:** c:\jfsrvr\CustomAgents\JetTracNotify\JetTracNotifyNom.ini @LogFileName @JobName  
 -#1@NumberPages -#2< variable 2> -#3< variable 3> -#4< variable 4> -#5< variable 5>  
 -#6< variable 6>  
**Comment:** JetTrac Notify Custom Agent by Pro Technology

4. Click OK to save the Task Definition.
5. In the task window, find the JFERROR job. Click on it and click on “Edit” on the right. Set up the Task Definition with the following information:

**Task ID:** JFERROR (Not changeable)  
**Program name:** java -jar c:\jfsrvr\CustomAgents\JetTracNotify\JTNotify.jar  
**Program options:** c:\jfsrvr\CustomAgents\JetTracNotify\JetTracNotifyError.ini @LogFileName @JobName  
 -#1@NumberPages -#2< variable 2> -#3< variable 3> -#4< variable 4> -#5< variable 5>  
 -#6< variable 6>  
**Comment:** JetTrac Notify Custom Agent by Pro Technology

6. Close out of the Task Definition window.

## 2.6 Configuring the Job in the Job Management Database

1. Now you must create a job step that calls the JetTrac Notify Affirmative task. This would be the last one of the job steps in the complete job you are setting up. To set up a new job, from JetForm Control main screen click File, Job Management Database, Jobs, then New Job. Define the rest of the steps that as needed. Under most circumstances, JetTrac Notify will be the last task in any JetForm Central job.
2. From the main screen of JetForm Control, click File, Job Management Database, Jobs. You will get a list of all jobs you have set up in JetForm Central. We will assume that you already have a job set up and you are adding an affirmative notification step. Click on the Job that you

The screenshot shows the 'Edit Job Step' dialog box with the following fields and values:

- Job name: NotifyTest
- Task: Notify
- Input file: \*
- Output file: \*
- On error: C
- Form file: \*
- Preamble file: \*
- Printer name: \*
- Macro number: 1
- Load flag: T
- Comments: \*

Buttons: Add Task..., Add Printer..., OK, Cancel

want to add notification to, then click “New Step”. The New Job Step screen will appear. The following are the fields that you need to fill in:

**Task ID** – from the drop down list, select JTNotify

**Input File** – leave this as \*

**Output file** – leave this as \*

**On Error** – Select S to Stop the Job if there is an error in JetTrac Notify

**Form File** – leave it as \*

**Preamble file** – This should be \* Delete whatever appears here.

**Printer Name** – leave it as \*

**Macro Number** – leave it at 1

**Load Flag** – leave it at T

**Comments** – type in a description like “Automatic Notification”

3. In order for JetForm Central to send an error notification, each job step in a job must have **On Error** set to **S** for stop. Otherwise, JetForm Central will continue on with the job. When **On Error** is set to stop, JetForm Central automatically launches JFERROR (and thus JetTrac Notify) when there is an error.
4. Close the job step window.
5. In the job definition window, select the JTNotify step and make sure that it is the last step in the job by putting it at the bottom of the list with the up and down arrow buttons in the window.
6. Close all windows, get back to the main JetForm Control screen and update the Job Management Database by clicking on Control, Reload Job Management DB. When asked if you want to save changes to the Job Management Database, click Yes.

## 3.0 RUNNING JETTRAC NOTIFY™

### 3.1 Submitting transaction files to JetForm Central

To submit a transaction file to JetForm Central, you place a data file into the Collector Directory. There are a number of ways to do this depending on the platform that you are running JetForm Central. You can FTP the file, use the JetForm Print Processor, use Named Pipes or simply copy the file to the Collector Directory.

### 3.2 How JetTrac Notify™ is triggered

As JetTrac Notify is set up as one of the steps of a job, whenever a job is triggered that needs to use JetTrac Notify, the processing happens automatically.

Alternatively, JetTrac Notify is triggered by JetForm Central when a job fails.

### 3.4 JetForm Central Log File

JetTrac Notify™ conforms to all JetForm requirements as a Custom Agent. It writes a status to the log file and creates the JETFORM.RSP. Here is a sample entry in the log file for a successful run of JetTrac Notify:

```
19991014121616 JetTrac Notify : processing completed
991014 12:16:16 C:\JFSRVR\jfsrvr.exe: [400]--> returned [0] JetTrac Notify : processing completed.
```

### 3.5 Batch Processing Output

When testing JetTrac Notify with a batch process, you should see output that looks something like this:

```
JetTrac Notify 1.0 - Generate Notification Action
(c) 2000, Pro Technology Automation Incorporated, Simi Valley, CA

----- Initialization -----
INI file: JTNotify.ini
Output file: job5.dat
Sending email to vectrant:dan@protechinc.com from dayl@protechinc.com b
job5 completed on 2000/02/10 07:25:52 -- Job ApCheck with input file Input.dat printed 35
pages on the printer \\protechnt\lexmark The job step was JFMERGE Test6
Connected to vectrant 25
Sent connect, waiting for 220
->got 220 protechinc.mail ESMTP service ready [1] (MDaemon v2.7 SP4 R)
Sent RSET, waiting for 250
->got 250 RSET? Well, OK.
Sent HELO, waiting for 250
->got 250 protechinc.mail Hello vectrant, pleased to meet you
Sent MAIL FROM:, waiting for 250
->got 250 <dayl@protechinc.com>, Sender accepted
Sent RCPT TO:, waiting for 250
->got 250 <dan@protechinc.com>, Recipient ok
Sent DATA, waiting for 354
->got 354 Enter mail, end with <CRLF>.<CRLF>
Sent finish, waiting for 250
->got 250 Ok, message saved
```

----- Processing Completed -----  
JetTrac Notify : processing completed