

Pro Technology Automation, Inc.
JetForm Central Development Toolkit
TDFAuditÔ
TDFCreateÔ
TranTracÔ

User's Guide

Version 1.1

September 10, 1999

REVISION HISTORY

Revision Number	Date	Comments
1.0	October 30, 1997	- Initial Beta release
1.01	November 20, 1997	- Added integration to JetForm Design Report With Fields
1.02	July 28, 1998	- Added documentation on TranTrac

User's Guide and Technical Documentation November 20, 1998

Copyright© Pro Technology Automation, Inc., 1997
All rights reserved.

Trademarks

JETFORM is a trademark of JETFORM Corporation.

TABLE OF CONTENTS

1. INTRODUCTION.....	4
1.1. PURPOSE	4
1.2. BACKGROUND.....	4
1.3. ORGANIZATION OF THIS DOCUMENT	4
2. INSTALLATION OF TDFAUDIT^Ô AND TDFCREATE^Ô	5
2.1. INSTALLATION PROCEDURE	5
2.2. TDFAUDIT TM EXAMPLE.....	3
2.3. TDFCREATE TM INSTRUCTIONS.....	11
2.4 TranTran TM	
Overview.....	1
1	

1. INTRODUCTION

1.1. Purpose

This document is a guide to using Pro Technology Automation's JetForm Central Development Toolkit, consisting of TDFAudit™, TDFCreate™ and TranTrac™. This document also includes instructions for installation and example files to help you see how the programs work.

1.2. Background

As JetForm Partners and end users develop JetForm Central applications, there are certain problems that can arise during the development and testing of TDF files (Transformation Definition Files). These three products arose out of many hours spent building TDF files and troubleshooting TDF files.

To illustrate the time savings here is a recent experience with the TDF tools. We delivered a custom JetForm Central training class at one of our clients in North Hollywood recently. One of their employees in Information Systems who was taking the class had been writing a TDF file the week before. This TDF file was over 3,000 records because there were over 1,000 fields he was processing. It took him over 2 days to write this TDF file.

One of the class exercises on the Transformation Agent covered the TDFCreate program. When this student told us what he had just done, I said let's try automatically creating his TDF file. We ran TDFCreate and it created a perfectly formatted TDF file in less than 10 seconds (the same TDF file took him over two days to write by hand!).

The other problem this same student had is that he had some errors in the TDF file and it took him almost 2 1/2 days to find the errors by hand. We ran his TDF file through the TDFAudit program, it found all his errors in less than 30 seconds!

1.3. Organization of this Document

This document is organized as follows:

Section 1: Introduction

Section 2: Installation Procedures for TDFAudit™ and TDFCreate™

Section 2.2: Explanation of example files for TDFAudit™.

Section 2.3: Explanation of example files for TDFCreate™.

2. INSTALLATION OF TDFAudit[®] and TDFCreate[®].

2.1. Installation Procedure

Installing these two products is very simple. There is one executable for each product. Simply copy the files to whatever folder you are using to develop TDF files and you will be ready to run.

2.2. The TDFAudit Example Files

- **Audit Tool for Central development**
- **Eliminates checking TDF file by hand for errors**
- **Compares TDF to form, field by field**

Before TDFAudit[™], you had to do a lot of work to find the errors in TDF files. TDFAudit[™] automatically checks for the following errors:

1. Duplicate names in the Extract sections.
2. It will find whether a field is in an Extract section but not a Script section and it will give a warning.
3. It will find whether a field is in the Script section but not the Extract section and it will give an error.
4. It will find whether a field is in the Extract or Script section but not in the field report (reports as warning, not fatal error).
5. Gives warning if in the script section the @fieldname did not match the ^field field-name or ^global fieldname.

We will look at an example to illustrate the functionality of TDFAudit[®].

The files used in this example are listed out here (you will be working from the MS-DOS command prompt):

Example.bat – batch file to run TDFAudit
Example.ifd – JetForm Design IFD file
Example.rpt – JetForm Design Report with Fields
Example.tdf – TDF file
Example.txt – output log file from running TDFAudit
TDFAudit.exe – executable program file

Example.bat file:

```
tdfaudit example.tdf >example.txt  
type example.txt
```

Example.rpt file:

Opening TDF input file example.tdf
Opening field report input file example.rpt

----- Processing TDF file -----

Found extract field <CUSTNUM>
Found extract field <CUSTNAME>
Found extract field <ADDRESS1>
Found extract field <ADDRESS2>
Found extract field <DATE>
Found extract field <AMOUNT>

----- Processing RPT file -----

----- Checking field use -----

----- Processing completed -----

Total of 6 fields processed
Total of 0 errors

Example.rpt file

Form Design: D:\PPTVIEW\JETFORM\Training\Exercise\TDFAudit\example.IFD
Created: 98/03/05 14:49
Last modified: 98/03/05 14:53

Form Name:
Description:
Version: 1.0
Creation Date:
Version Date:
Designer:
Contact:

Printer: PCL Printer JetForm version: 402

... DocVars ...
"MessageType" ""
"JfSymbolSet" "Yes,1252,437"
"JfDesignVersion" "Version 5.1.281"

Database Rules: none

User Macros : none

----- Page: 1:'1' -----

Page Rules : none

Field:CUSTNUM [1] Lines: 1 Characters: 10 Angle: 0

Field:CUSTNAME [1] Lines: 1 Characters: 30 Angle: 0

Field:ADDRESS1 [1] Lines: 1 Characters: 30 Angle: 0

Field:ADDRESS2 [1] Lines: 1 Characters: 30 Angle: 0

Field:DATE [1] Lines: 1 Characters: 10 Angle: 0

Field:AMOUNT [1] Lines: 1 Characters: 12 Angle: 0

The TDF file used in this example is as follows:

O " N 1 N N Y N C|

E CUSTNUM
E CUSTNAME
E ADDRESS1
E ADDRESS2
E DATE
E AMOUNT

#startscript *
^field CUSTNUM
@CUSTNUM
^field CUSTNAME
@CUSTNAME
^field ADDRESS1
@ADDRESS1
^field ADDRESS2
@ADDRESS2
^field DATE
@DATE
^field AMOUNT
@AMOUNT
#endscript

Detailed example of errors that TDFAudit[®] can find:

1. From Windows Explorer, double click on c:\tdfaudit\example.ifd. JetForm Design will launch and display the form used in this example.
2. Click File, Report With Fields, then do a Save As c:\tdfaudit\example.rpt
3. Pull up an MS-DOS prompt to c:\tdfaudit and type EXAMPLE. This is a batch file that simply runs TDFAudit, then lists out the error file. The command line is:

```
tdfaudit example.tdf >example.txt  
type example.txt
```

4. There should not be any errors reported.

5. Edit the example.tdf file. In the E CUSTNUM line, take out the M in CUSTNUM. Save the file, then type EXAMPLE. This is the error file generated:

*** Contents of Error file ***

```
Opening TDF input file example.tdf  
Opening field report input file example.rpt
```

```
----- Processing TDF file -----
```

```
Found extract field <CUSTNU>
```

```
Found extract field <CUSTNAME>
```

```
Found extract field <ADDRESS1>
```

```
Found extract field <ADDRESS2>
```

```
Found extract field <DATE>
```

```
Found extract field <AMOUNT>
```

```
** Warning: script name not found in extract section <CUSTNUM>
```

```
----- Processing RPT file -----
```

```
** Warning: field in form report but NOT in TDF file <CUSTNUM>
```

```
----- Checking field use -----
```

```
** Warning: field in Extract section but not in script section <CUSTNU>
```

```
** Warning: field in Extract section but not in field report <CUSTNU>
```

```
----- Processing completed -----
```

```
Total of 6 fields processed
```

```
Total of 4 errors
```

```
*** End of Error file ***
```

Explanation of Errors:

TDFAudit has identified a field in the Script (CUSTNUM) that is not found in the Extract, which is a problem. This is because of a typo in the Extract field name CUSTNUM. It is also saying Warning: field in form report but NOT in TDF file <CUSTNUM>, which is not a fatal error, but you may want to know it. It also reports that there is a field in the Extract Section but not in the Script section or in the field report.

6. Edit the example.tdf again and fix the first error (in the Extract Section, put the M back in CUSTNUM).

In the Script section, where it says @ADDRESS1, change it to @ADDRES1 (drop the second S). Save the file, then type EXAMPLE and look at the errors generated.

*** Contents of Error file ***

Opening TDF input file example.tdf

Opening field report input file example.rpt

----- Processing TDF file -----

Found extract field <CUSTNUM>

Found extract field <CUSTNAME>

Found extract field <ADDRESS1>

Found extract field <ADDRESS2>

Found extract field <DATE>

Found extract field <AMOUNT>

** Warning: script field mismatch <@ADDRESS1

>

** Warning: script name not found in extract section <ADDRES1>

----- Processing RPT file -----

----- Checking field use -----

** Warning: field in Extract section but not in script section <ADDRESS1>

----- Processing completed -----

Total of 6 fields processed

Total of 3 errors

*** End of Error file ***

Explanation of Errors:

It says "Warning: script field mismatch <@ADDRESS1>". This means that there is no matching ^field ADDRES1 that it found. It also warns you that the "script name not found in extract section <ADDRES1>", which is a fatal error. It also says that "field in Extract section but not in script section <ADDRESS1>", which is a warning, but not fatal.

Note: You do not have to match the field names in the two script lines together, but it avoids confusion with what field names are used where. In our development, we use the same field name in the Extract section and the Script.

7. The next step is to change a field name in the form so it doesn't match the TDF file. Go into JetForm Design and change the DATE field to DTE. Click File, Report with Fields, then do a Save As c:\tdfaudit\example.rpt. Type example and looks at the errors generated.

*** Contents of Error file ***

Opening TDF input file example.tdf
Opening field report input file example.rpt

----- Processing TDF file -----

Found extract field <CUSTNUM>
Found extract field <CUSTNAME>
Found extract field <ADDRESS1>
Found extract field <ADDRESS2>
Found extract field <DATE>
Found extract field <AMOUNT>

----- Processing RPT file -----

** Warning: field in form report but NOT in TDF file <DTE>

----- Checking field use -----

** Warning: field in Extract section but not in field report <DATE>

----- Processing completed -----

Total of 6 fields processed
Total of 2 errors

*** End of Error file ***

Explanation of Errors:

This is saying that there is a field DTE in the form but not referenced in the TDF file. This is a warning, but not fatal. It also says that there is a field DATE in the Extract section but not in the form. This is also a warning.

2.3. TDFCreate Instructions

To create a TDF file for a delimited ASCII file, you create a text file of field names, like the following example (example file provided is Tdfcreat.txt):

```
custnum  
lastname  
firstname  
street  
city  
state  
zip  
date  
desc1  
desc2
```

At the MS-DOS command prompt, type the following:

```
TDFCREAT textfilename (for example TDFCREAT fields.txt)
```

Where TDFCREAT is the executable program (TDFCREAT.EXE) and textfilename is the name of the text file containing the field names. A TDF file will be created with the same filename but with a TDF extension. The TDF file created for the above fields is:

```
O " N 1 N N Y Y C|
```

```
E custnum  
E lastname  
E firstname  
E street  
E city  
E state  
E zip  
E date  
E desc1  
E desc2
```

```
#startscript *  
^field custnum  
@custnum  
^field lastname  
@lastname  
^field firstname
```

```
@firstname
^field street
@street
^field city
@city
^field state
@state
^field zip
@zip
^field date
@date
^field desc1
@desc1
^field desc2
@desc2
#endscript
```

You can modify the TDF file any way you want for a different delimiter, make ^global fields instead of ^field, add ^form and/or ^eject commands, etc.

Now you can run the newly created TDF file against the form report with fields using TDFAudit to find any errors (see the example on page 5 starting at step 3. for instruction).

2.4 TranTrac

TranTrac was developed to get around a limitation that the JetForm Central Transformation Agent has in creating a TDF file for an ASCII delimited file that is larger than 4096 bytes. TranTrac is run as a custom agent instead of the JetForm Central Transformation Agent (JFTRANS) prior to running the Calculation Agent (JFCALC) and/or the Print Agent (JFMERGE).

The input file is a delimited ASCII file. Here is an example of a partial file:

```
CoBorr Assets=|CoBorr Liabilities=|VA=|Conventional=|Other=|FHA=|X|FmHA=|(specify)=|
|Agency Case Number=|07645|Lender Case Number=|9800000556
```

The developer still creates a TDF file using TDFCreate and validates there are no errors with TDFAudit. Here is an example of the TDF file for the above file:

```
O " N 1 N N Y Y C|
```

```
E 1stcheckbox
E CBCOB_ASIN
```

E 2ndcheckbox
 E CBCOB_LIAB
 E va
 E CBVA
 E conventional
 E CBCONV
 E other
 E CBTDTYPEOTH
 E fha
 E CBFHA
 E fmha
 E CBFMHA
 E otherexplain
 E VTDOTHEXPLN
 E agcaseno
 E VAG_CASENUM_A
 E lecaseno
 E VLOAN_NUM_A

(Note: we have omitted the script section in this example)

The command line in the Task to run TranTrac as a Custom Agent in JetForm Central is:

```
trantrac inputfile.dat outputfile.fnf TDFfile.tdf 1 file.post "^field"
```

Here is an explanation of each component:

Trantrac	Executable filename, should be fully qualified with drive and path
inputfile.dat	Input data file in ASCII delimited format
outputfile.fnf	New output file that will be created
TDFfile.tdf	TDF file that will be used during transformation
1	Number of records to skip in input file (usually to skip job card)
file.post	ASCII file that contains data you want to append to end of output file*
"^field"	Specifies whether you want ^field or ^global in output file

* It may be necessary to append data at the end of the Field Nominated output file, particularly if you want to call specific forms. Here is an example of the output file:

```

^global 1stcheckbox
CoBorr Assets=
^global CBCOB_ASIN

^global 2ndcheckbox
  
```

CoBorr Liabilities=

^global CBCOB_LIAB

^global va

VA=

^global CBVA

^global conventional

Conventional=

^global CBCONV

^global other

Other=

^global CBTDTYPEOTH

^global fha

FHA=

^global CBFHA

X

^global fmha

FmHA=

^global CBFMHA

^global otherexplain

(specify)=

^global VTDOTHEXPLN

^global agcaseno

Agency Case Number=

^global VAG_CASENUM_A

07645

^global lecaseno

Lender Case Number=

^global VLOAN_NUM_A

9800000556

^page 1

^page 2

^page 3

^page 4

Note: This output file would be merged with the form and pages 1,2,3,4 of the form would be printed with the above data.