

User Instructions

Ebase Data Transfer Tool, V1.0

(FM3/4, V1.0.2 Version)

Larry Bednar

February 4, 2003

Purpose

This document explains the use of the “Ebase data transfer tool”.

Overview

The “Ebase data transfer tool” (EDTT) is designed to assist users in “migration” of data from a damaged copy of Ebase V1.0.0 and V1.0.2 database files to a fresh copy of “standard” Ebase V1.0.2 database files.

If this data migration is performed manually, a very lengthy series of manual steps (button clicks, menu selections, responses to dialog windows, etc) is required to complete the job correctly. Because of the large number of steps, even an expert user can easily overlook some essential task, hit the wrong button, etc. The process is thus inherently error-prone.

Many of the steps required for “data migration” are readily scripted by FileMaker, and are therefore very amenable to automation. Automation of these steps results in a substantially reduced chance of “operator error” during data migration. Automated steps, once set up correctly, are much more likely to be implemented correctly than steps in a “manual” migration.

EDTT also provides a checklist through which a user can easily keep track of their progress through the data migration process. If the work session is interrupted, the user can much more easily return and carry out the correct steps in sequence to complete the process already started. Multiple data migrations may be tracked in the database, each one represented by a separate record in the provided FileMaker database provided to run the application.

The version of the EDTT documented here is designed to transfer data from damaged Ebase V1.0.x files to a clean set of Ebase V1.0.2 database files. Because EDTT requires 3 “non-standard” Ebase scripts in each V1.0.2 file targeted for data import, copies of those Ebase files are provided with EDTT.

Requirements

The following requirements should assure that the EDTT application works correctly:

1. A Windows-based PC must be used.
2. A copy of FileMaker Pro V5 must be installed on the PC where the data transfer is being attempted.
3. The data to be transferred to new databases must be in Ebase V1.0.x database files

At present, the tool cannot be guaranteed to perform correctly in environments that do not provide these features. However, there is one report that the version developed for FileMaker

V5/Ebase V1.0.3 has been successfully used by a long-experienced Ebase developer using a Macintosh. He reported that it was very nice to have the transfer more fully automated.

Components

EDTT is provided in a WinZip self-extracting archive. This file contains:

1. A database (“edtt_data_transfer_scripts.fp3”) containing the checklist of data migration tasks and scripts designed to automate data migration tasks.
2. A database (“edtt_custom_objects.fp3”) containing scripts designed to create descriptions of fields, layouts, and scripts that will be orphaned by the data migration unless the user recreates them in the new Ebase database files.

[NOTE: This set of functions is not yet completed/functioning fully...]

3. A set of Ebase V1.0.2 files which is modified slightly. Most of these files contain 2 “data import” scripts designed to import data from the corresponding V1.0.x database.

Installation

To install and use the tool, the following steps are followed:

1. The user receives a self-extracting WinZip archive file that contains the EDTT application.
2. The user places the EDTT archive file in the parent folder of the folder that currently holds the Ebase database files.
3. The user extracts the files from the archive. This creates a new folders named "edtt_new" and “edtt_old” which are in the same “parent” folder that holds the current Ebase files.

The “edtt_new” folder contains copies of all “standard” Ebase V1.0.2 database files, modified to work with the “data transfer tool”.

The folder also contains:

- The “edtt_data_transfer_scripts.fp3” database. (Contains master scripts for running imports and checks for orphaned objects in the old copy of Ebase. Also provides a checklist of data migration steps, and can be utilized to track progress through one or more data migration processes.)
- The “edtt_custom_objects.fp3” database. (Contains scripts for checking old Ebase files for fields, scripts and layouts that will be orphaned unless the user manually adds them to the new Ebase files. After these scripts are run, the database contains a record of each these “orphan objects”.)

Preparation for Use

1. The master password to the old and new databases should be set to the same value

Several EDTT scripts prompt the user for passwords before opening Ebase database files. In some cases, the prompt window for the password is not sufficient to distinguish old and new databases. (If for instance, you’re transferring data from a corrupted version of NAMES_.102 into a new copy of NAMES_.102.) Because of this, it is recommended

that the “master” passwords for both the old and new databases be set to the same value before the data transfer process is started.

Instructions for Use

1. The user places copies of the current Ebase databases in the "edtt_old" folder.

The "data transfer tool" utilizes these database copies and the newly created files in the “edtt_new” folder exclusively. As a result, there should be no danger whatsoever to your original database files.

(If the PC has a copy of Perl installed, the included script “make_ebase_copies.pl” can be run to perform this task. This script will prompt the user for the name of the folder containing the current Ebase database files. The script will determine whether Ebase V1.0.0, 1.0.2 or 1.0.3 is installed in that folder and will move files carrying the standard Ebase extensions, “100”, “102”, or “103”).

[NOTE: The perl scripts mentioned above have not been tested with the FM3 version of EDTT yet. Although there shouldn’t be a problem...]

2. The user opens the database, creates a new record representing a new "data transfer" attempt, and follows the instructions on the appropriate layout (Figure 1). The user stops at the “Review ‘edtt_custom_objects.fp3’ database...” step.

The user should be sure to choose the correct layout – three versions are provided which run different scripts:

V1.0.2 to V1.0.2 transfer steps

V.1.0.0 to V1.0.2 transfer steps

After completing each task, the user "checks off" the task, so the database provides a record of which steps have been completed. As a result, the process can be easily continued after an interruption.

edtt_data_transfer_scripts.FP3

Ebase Data Transfer Steps FM 3
V1.0.2 to V1.0.2

This database and layout are designed to assist in moving data from a corrupted or damaged copy of Ebase V1.0.2 into a newly installed copy of Ebase V1.0.2

TransferID: 4
CreationDate: 1/20/2003
DatabaseDesc:

Set "completed" to blank
Run data transfer scripts
Calc next seial numbers
Assign next seial numbers

NextACTIONSActionNumber: 1
NextCONTACTContactNumber: 39
NextNAMESRecordNumber: 1
NextNOTESNoteNumber:
NextPAYMENTSReceiptNumber: 15
NextPLEDGESPledgeKey:
NextSTAFFStaffIDNumber: 1

Completed

Step	Description	Completed
Step01	Create folder 'edtt_old' in parent folder for Ebase data folder	<input type="checkbox"/> Y
Step02	Place a copy of source Ebase files in 'edtt_old' folder	<input type="checkbox"/> Y
Step03	Open 'edtt_old\CUSTOM_' database and 'find all' records	<input type="checkbox"/> Y
Step04	Open 'edtt_old\DOCUMNT_' database and 'fin all' records	<input type="checkbox"/> Y
Step05	Open 'edtt_old\PROFILE_' database and 'find all' records	<input type="checkbox"/> Y
Step06	Open 'edtt_old\PROFILE_' database and 'find all' records	<input type="checkbox"/> Y
Step07	Open 'edtt_old\SETUP_' database and 'find all' records	<input type="checkbox"/> Y
Step08	Open 'edtt_old\SRCDEFS_' database and 'find all' records	<input type="checkbox"/> Y
Step09	Open 'edtt_old\STAFF_' database and 'find all' records	<input type="checkbox"/> Y
Step10	Open 'edtt_old\ZIP_' database and 'find all' records	<input type="checkbox"/> Y
Step11	Open 'edtt_old\ZIPCODE_' database and 'find all' records	<input type="checkbox"/> Y
Step12	Close copy of Ebase in 'edtt_old' folder	<input type="checkbox"/> Y
Step13		<input type="checkbox"/> Y
Step14	Create copy of 'edtt_old\actions_' as 'edtt_old\actions_old'	<input type="checkbox"/> Y
Step15	Create copy of 'edtt_old\names_' as 'edtt_old\names_old'	<input type="checkbox"/> Y

Figure 1. "Data transfer" layout in "edtt_data_transfer_scripts.fp3" database

The user will be prompted for Ebase passwords as EDTT scripts open Ebase database files. The user should enter the Ebase "master" password when prompted for passwords.

3. In the new Ebase files provided by EDTT, the user creates fields matching custom fields in the source databases.

It is important to create the new fields with exactly the same names and characteristics as in the old Ebase files. If done correctly, the data transfer steps will then import custom data into the new Ebase files.

4. The user continues with the remaining data transfer steps.
(In a case where custom fields are to be imported, the user's specification of import by "matching fields" is necessary. Because of this, the data import dialogs are displayed if the user indicates that custom fields are to be imported.)
5. The user checks the contents of the new database files closely to verify that data has transferred correctly.
6. The user imports scripts and layouts developed during customizations of the old database files.

Imported scripts and layouts must be checked closely to verify that field, script, and layout references are correct.

7. The user recreates any layout features or modifications of standard Ebase scripts developed during customization of the old database files, and which cannot be implemented through the “import” operations.

(This step may require some careful thinking and planning.)

Databases checked for custom objects [Not yet functioning]

Scripts in the “edtt_custom_objects.fp3” database will check for custom objects in the following databases:

- actions_.10x
- contact_.10x
- custom_.10x
- groups_.10x
- invoices_.10x (only for V1.0.2 data sources)
- names_.10x
- notes_.10x
- paymnts_.10x
- pledges_.10x
- profile_.10x
- setup_.10x
- solicit_.10x
- srcdefs_.10x
- staff_.10x
- summary_.10x
- zip_.10x
- zipcode_.10x

These scripts identify any fields, layouts or scripts that do not have counterparts in the new Ebase databases.

Transferred Data

The “Ebase data transfer tool” should import data into the following Ebase database files from the corresponding source databases::

- actions_.102
- contact_.102
- custom_.102
- deleted_.102
- document_.102
- groups_.102
- names_.102
- notes_.102
- paymnts_.102
- pledges_.102
- profile_.102

setup_.102
solicit_.102
srcdefs_.102
staff_.102
summary_.102
zip_.102
zipcode_.102

Assistance and Use Agreement

The application is provided for use free of charge, and the user is free to distribute or modify the tool under the following conditions:

1. The distributed product must retain mention of the original author (available below) and contact information for the author.
2. Distributions of modified versions of the product must indicate that they have been modified, who performed the modification, and when the modifications were implemented.
3. Those distributing the tool must not license it, register it, or in any other way restrict the distribution of the tool by others. You got it from me without charge and with the right to modify and distribute – those who get it from you should have the same rights.

The user accepts full responsibility for evaluating and preventing problems that might arise from use of the application. The author cannot guarantee that the tool will perform correctly in every possible environment.

Copies of the tool and/or assistance in use or modification may be obtained from the author:

Larry Bednar
Bednar Consulting
3226 NE 24th Ave.
Portland, OR 97212
503.493.8542
lbstat@easystreet.com

The author welcomes and encourages feedback and information about use and modification of the tool, which will be used for continued improvement of the product.

Due to time restrictions, suggestions for modification will be triaged and prioritized together with the author's other ongoing projects. As a result, even suggestions of high value and high ease of implementation may *not* be implemented immediately. However, the author will maintain a formal record of problem reports/suggestions to ensure that comments and ideas for product improvement will not be accidentally lost.