
# IMPEX for SWORD & TelcoMgr

# IMPEX for SWORD & TelcoMgr

Import & Export Utility Program

***By ABCI***

# Table of Contents

# 1 Introduction to IMPEX for SWORD and TelcoMgr

**Optional File Import and Export Utility For SWORD™ and TelcoMgr™ Standard or Professional Editions**

**IMPEX for SWORD** is a database file utility designed for the purpose of importing or exporting data used by SWORD:ServiceBooks™ and TelcoMgr™.

**Designed for Advanced Users**
We strongly recommend that you first become familiar with **SWORD™ and/or TelcoMgr™** and the various tables that are used before attempting to use **IMPEX** for the first time.

SWORD: Service Books Import & Export Options

- Export Files
- Import Files
- Browse Imported SWORD Files


TelcoMgr Import & Export Options

- Advanced Import
- Importing CSV Files
- Importing Excel Files
- Advanced Export
- Exporting as CSV Files
- Exporting as HTML Files
  Exporting as Excel
- Filtering Exports
- Simple Tagging
- Global Search and Tag
- Query By Example
- Field Data Manager
- File Structures for Importing

**Professional Services** for data IMPORT and EXPORT of TelcoMgr files are also available from ABCI Software.

## 1.1 About ABCI Software

# Contact Us

**ACCESS BUSINESS COMMUNICATIONS, INC.**
16835-236 Algonquin Street
HUNTINGTON BSurf City USAEACH, CA 92649

**phn (800) 675-2415**
**fax (714) 442-9994**

**Hours of Operation**

Monday - Friday 09:00 a.m. - 05:00 p.m. PST

Send to: marketing@abci-software.info

We have created a virtual or office anywhere using Internet technology for voice, video and data transactions, which has dramatically reduced our carbon foot-print.

Call center answering services provided by Ansafone Communications Company Communications Co. Packaging and shipping products provided by UPAK and UpaknShip, available online at Yahoo! Other resources are available from our online directory.








## 1.2 EULA

ACCESS BUSINESS COMMUNICATIONS, INC. (ABCI)
ELECTRONIC END USER LICENSE AGREEMENT
FOR IMPEX FOR SWORD, IMPEX FOR TELCOMGR

**NOTICE TO USER:**

THIS IS A CONTRACT. BY INSTALLING THIS SOFTWARE YOU ACCEPT ALL THE TERMS AND
CONDITIONS OF THIS AGREEMENT.

This ACCESS BUSINESS COMMUNICATIONS, INC. ("ABCI") End User License Agreement accompanies the IMPEX FOR SWORD or IMPEX FOR TELCOMGR products (hereafter IMPEX) and related explanatory materials "Software"). The term "Software" also shall include any upgrades, modified versions or updates of the Software licensed to you by ABCI. Please read this Agreement carefully. At the end, you will be asked to accept this agreement and continue to install or, if you do not wish to accept this Agreement, to decline this agreement, in which case you will not be able to use the Software.

Upon your acceptance of this Agreement, ABCI grants to you a nonexclusive license to use the Software, provided that you agree to the following:

**1. Use of the Software**. Simply stated, each single user licenses grants the user permission to use one copy of IMPEX on one single hard drive.

- You may install the Software on a hard disk or other storage device; install and use the Software on a file server for use on a network for the purposes of (i) permanent installation onto hard disks or other storage devices or (ii) use of the Software over such network; and make backup copies of the Software.
- You may not make and distribute copies of the Software, including copies for commercial distribution.

**2. Copyright and Trademark Rights.** The Software is owned by ABCI and its suppliers, and its structure, organization and code are the valuable trade secrets of ABCI and its suppliers. The Software also is protected by United States Copyright Law and International Treaty provisions. You may use trademarks only insofar as required to comply with Section 1 of this Agreement and to identify printed output produced by the Software, in accordance with accepted trademark practice, including identification of trademark owner's name. Such use of any trademark does not give you any rights of ownership in that trademark. Except as stated above, this Agreement does not grant you any intellectual property rights in the Software.

**3. Restrictions**. You agree not to modify, adapt, translate, reverse engineer, decompile, disassemble or otherwise attempt to discover the source code of the Software. Although you may customize the installer for the Software as documented on

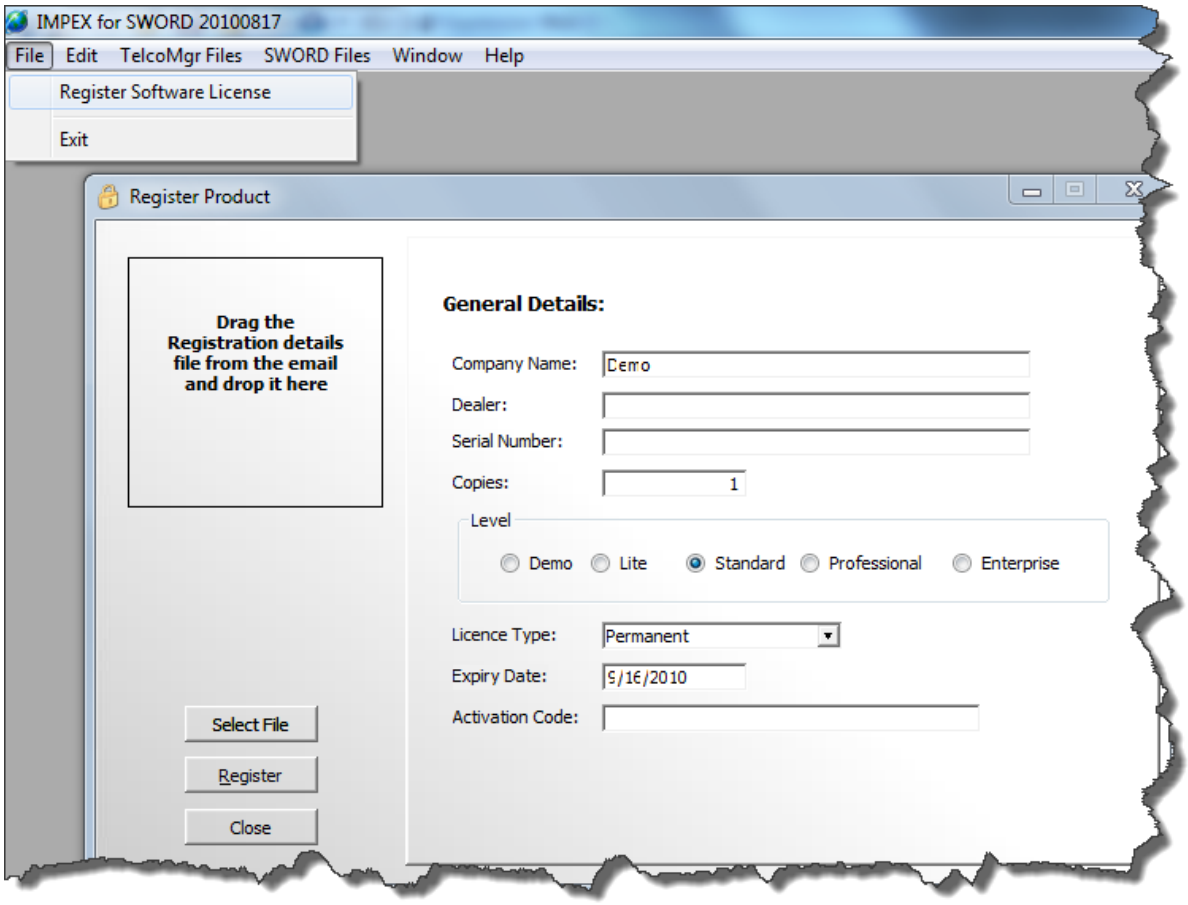
the ABCI IMPEX CD-ROM or web site download file (e.g.,installation of additional help files), you may not otherwise alter or modify the installer program or create a new installer for the Software.

**4. No Warranty.** The Software is being delivered to you AS IS and ABCI makes no warranty as to its use or performance. ABCI and its suppliers do not and Cannot warrant the performance or results you may obtain by using the software Or documentation. ABCI and its suppliers make no warranties, express or Implied, as to noninfringement of third party rights, merchantability, or Fitness for any particular purpose. In no event will ABCI or its suppliers be Liable to you for any consequential, incidental or special damages, including Any lost profits or lost savings, even if an ABCI representative has been Advised of the possibility of such damages, or for any claim by any third Party. Some states or jurisdictions do not allow the exclusion or limitation of incidental, consequential or special damages, or the exclusion of implied warranties or limitations on how long an implied warranty may last, so the above limitations may not apply to you.

**5. Governing Law and General Provisions**. This Agreement will be governed by the laws of the State of California, U.S.A., excluding the application of its conflicts of law rules. This Agreement will not be governed by the United Nations Convention on Contracts for the International Sale of Goods, the application of which is expressly excluded. If any part of this Agreement is found void and unenforceable, it will not affect the validity of the balance of the Agreement, which shall remain valid and enforceable according to its terms.
You agree that the Software will not be shipped, transferred or exported into any country or used in any manner prohibited by the United States Export Administration Act or any other export laws, restrictions or regulations. This Agreement shall automatically terminate upon failure by you to comply with its terms. This Agreement may only be modified in writing signed by an authorized officer of ABCI.

**6. Notice to Government End Users.** The Software and Documentation are “Commercial Items," as that term is defined at 48 C.F.R. §2.101, consisting of "Commercial Computer Software" and "Commercial Computer Software Documentation," as such terms are used in 48 C.F.R. §12.212 or 48 C.F.R. §227.7202, as applicable. Consistent with 48 C.F.R. §12.212 or 48 C.F.R. §§227.7202-1 through 227.7202-4, as applicable, the Commercial Computer Software and Commercial Computer Software Documentation are being licensed to U.S. Government end users (A) only as Commercial Items and (B) with only those rights as are granted to all other end users pursuant to the terms and conditions herein. Unpublished-rights reserved under the copyright laws of the United States.

## 1.3 Registering IMPEX License

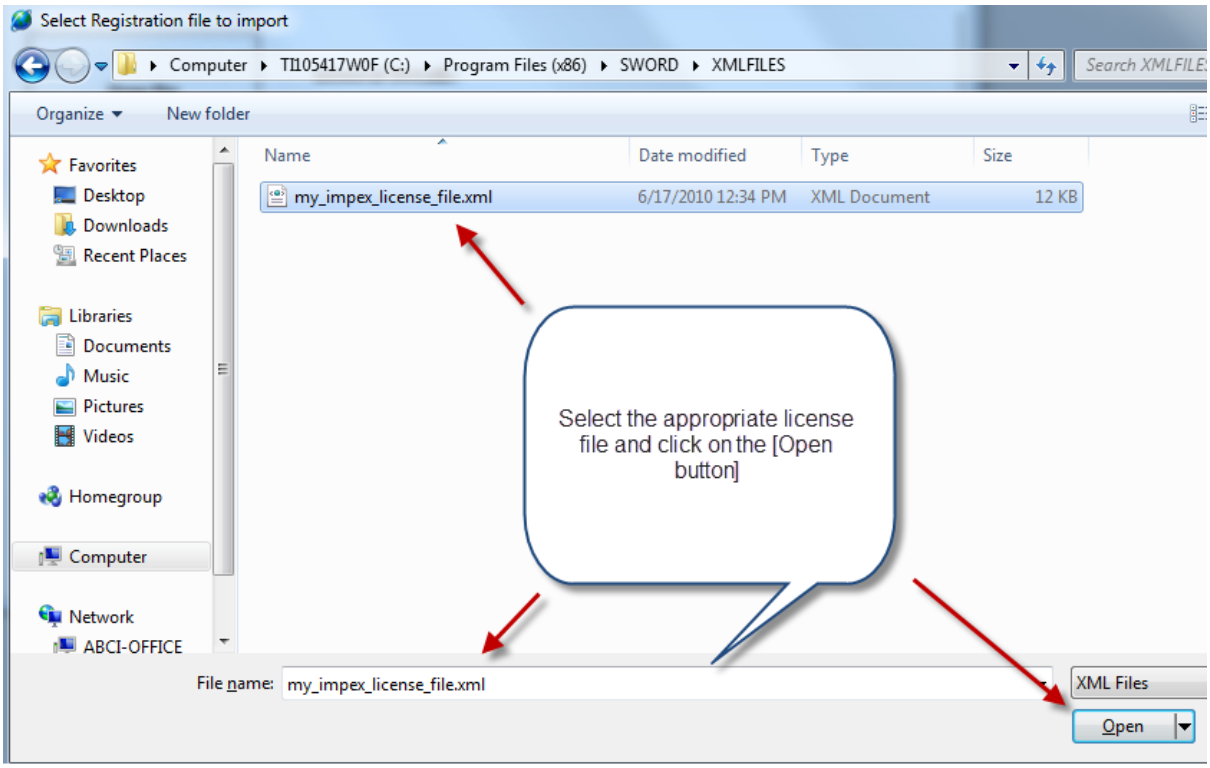



To Register the End-User-License for IMPEX you may keyboard the information into the Register Product window, copy and paste the information field by field or load the XML license file (if available).

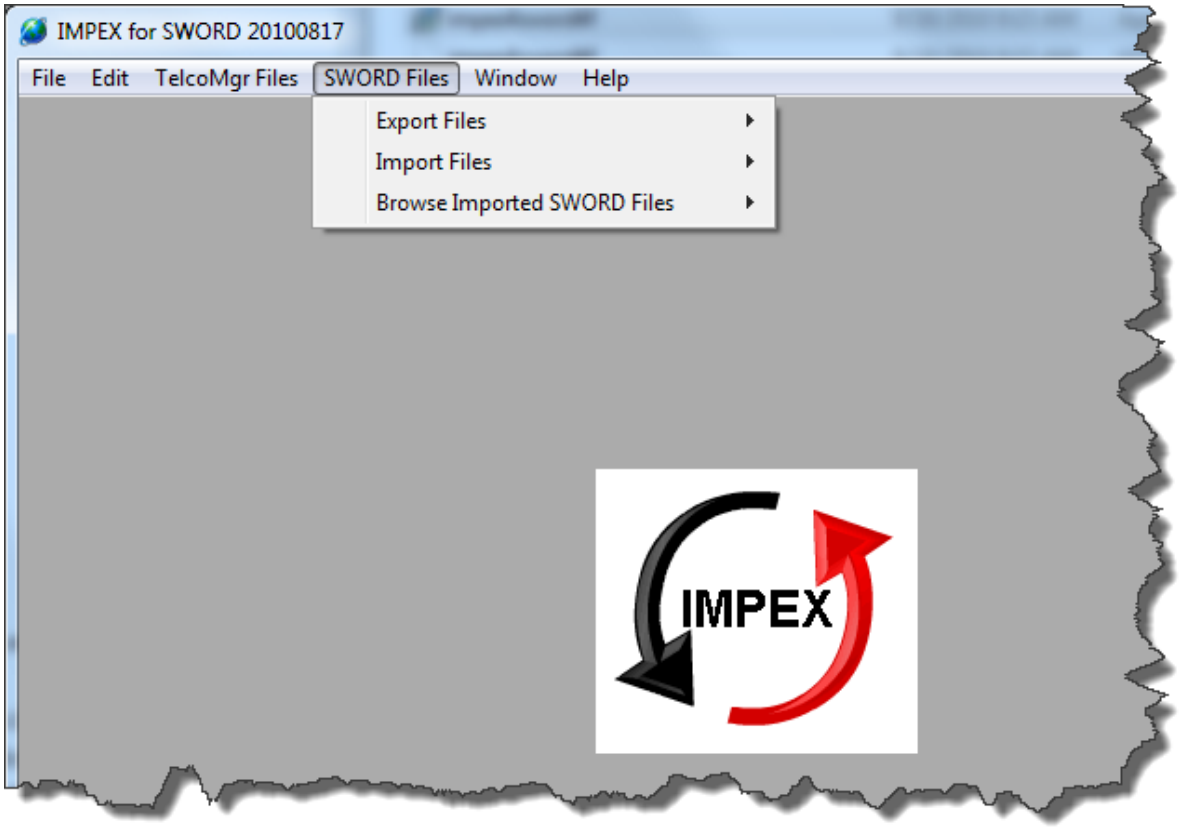
**User Tip:** If you keyboard the information it is very important that you match the case and exact spelling of the license information provided to you, which is typically sent inside an email message. If you copy and paste the information, make certain that you are not adding any additional characters or spaces (before or after).

### Loading the XML License file

1. Typically the XML license type file is included as an attachment in an email to the registering user.
2. Inside the email save the XML attachment to your computers desktop or to another convenient file space.
3. Click on the [**Select File button**] and navigate to the XML license file that was saved above.
4. Click the Open File dialog button.
5. Click on the [**Register button**] inside the **IMPEX Register Product window**.



# 2 SWORD ServiceBooks Import & Export Options




## Export Files

- Chart of Accounts
- Contacts
- Customers
- Items

## Import Files

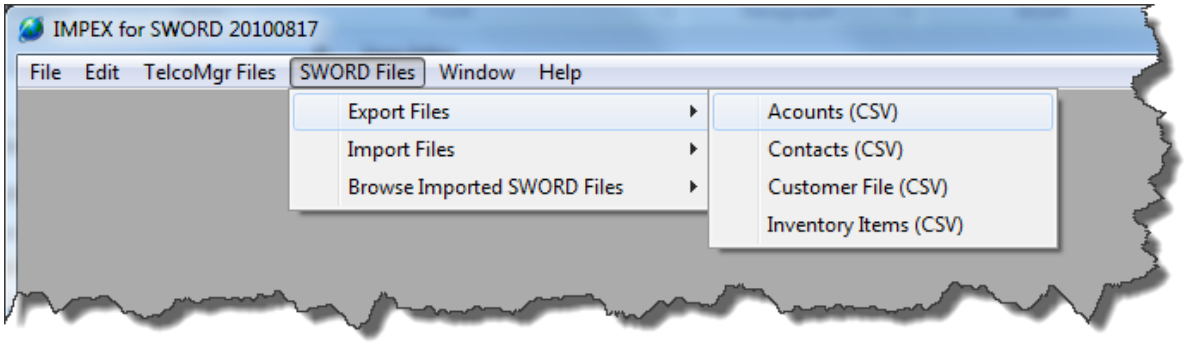
- Chart of Accounts
- Contacts File
- Customers
- Inventory Items

## Browse Imported SWORD Files

- Contact Names
- Chart of Accounts
- Customers

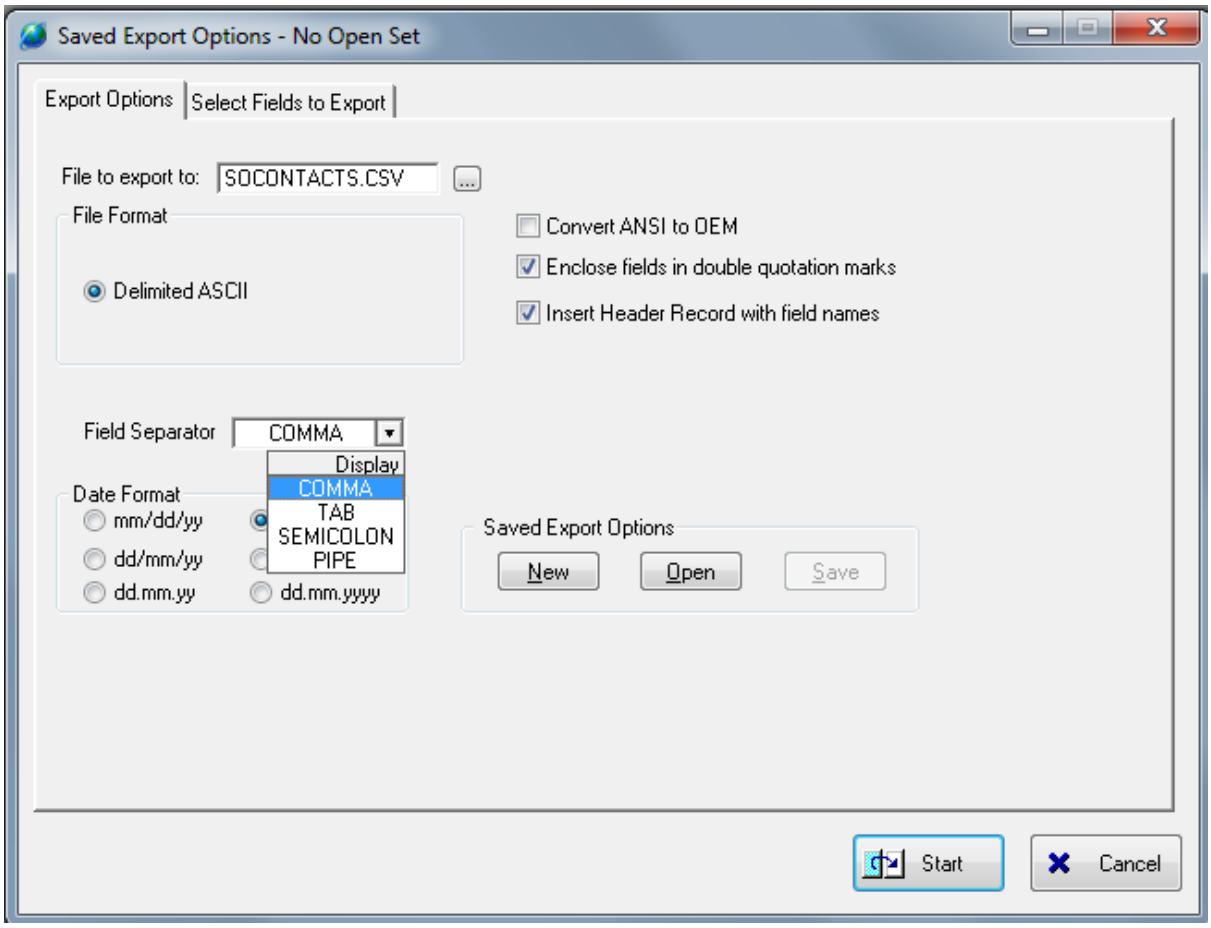
- Inventory Items

## 2.1 Export SWORD Files




- Chart of Accounts
- Contacts
- Customers
- Items

The above listed SWORD files may be exported as Comma Separated Values (CSV).

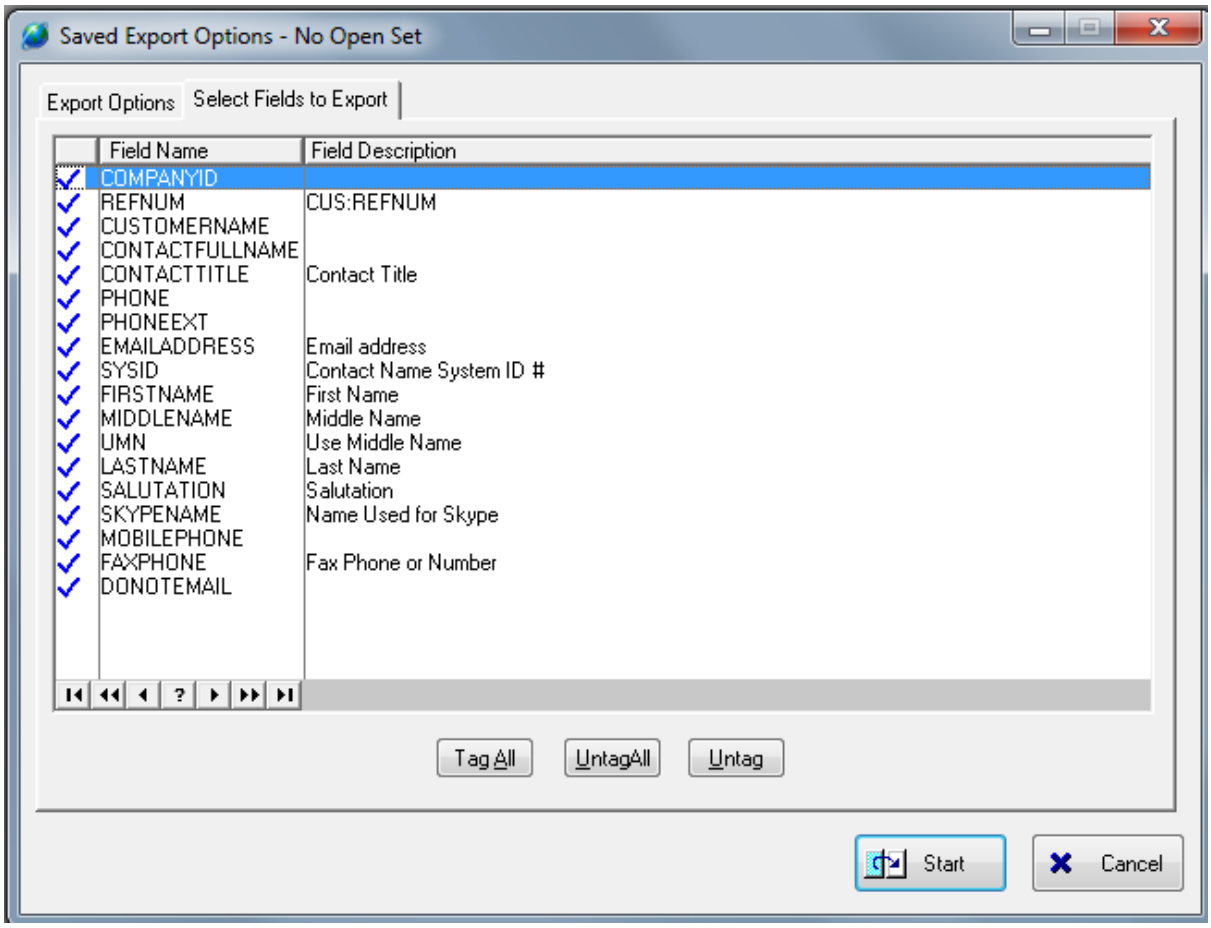



Alternately, the file may be delimited with a TAB, SEMICOLON or PIPE symbol.

**Export Options**

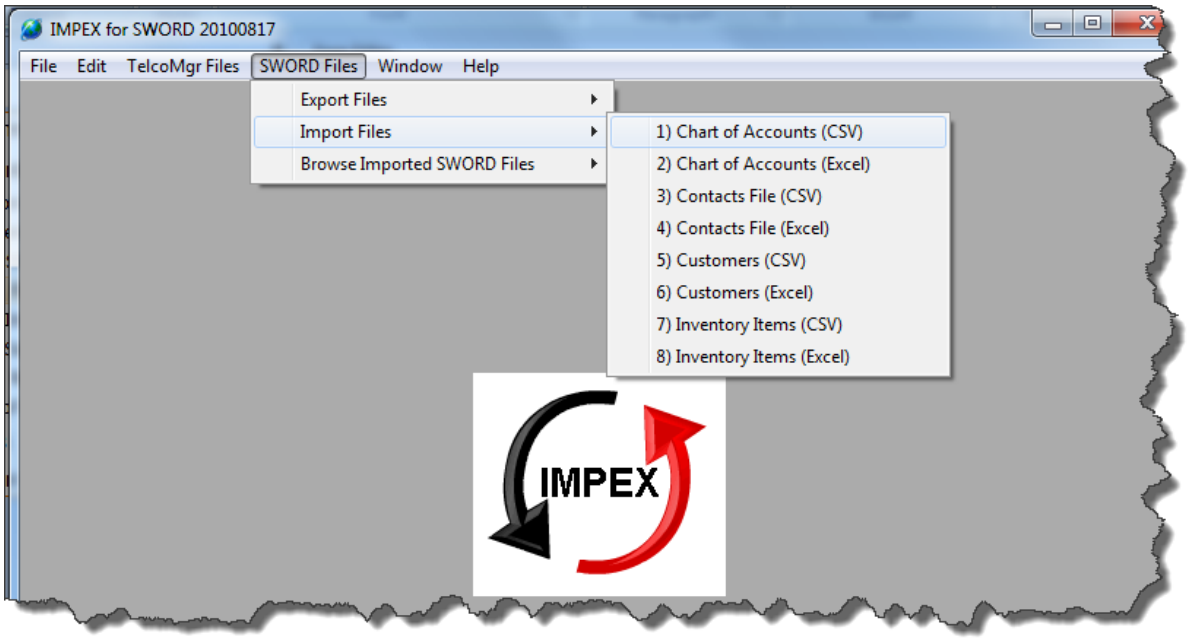
- ❑ **File to export to:** > The default file name is shown, however select the [...] lookup button to navigate to a different file or enter the name of a new file.
- ❑ **Convert ANSI to OEM** > Windows programs typically use a different character set than do DOS programs. The DOS character set is known as the DOS/OEM character set, and includes various line drawing characters and miscellaneous characters not in the Windows/ANSI set. The Windows/ANSI character set includes many accented characters not in the DOS/OEM character set.
- ❑ **Enclose fields in double quotations marks** > Allows fields with COMMAS to be encapsulated as a single field of data.
- ❑ **Insert Header Record with field names** > The SWORD field names are exported on the first/header record.
- ❑ **Save Export Options** > The options selected may be saved and reused each time an export is executed.

**Select Fields to Export**




One or more of the fields may be exported by selecting (check) the fields that you intend to export.

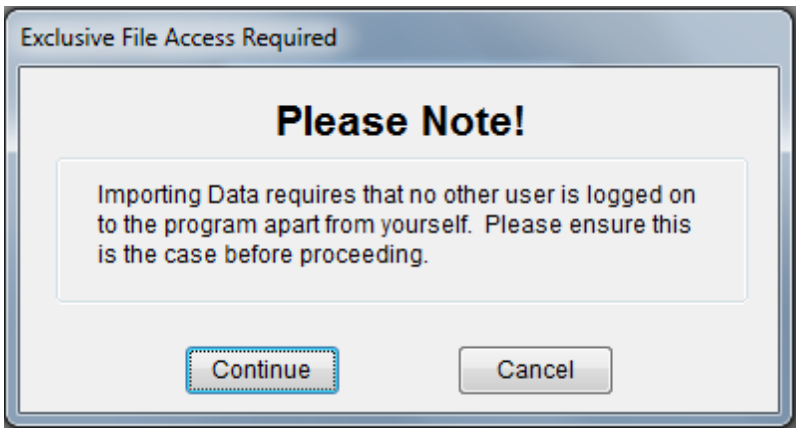
## 2.2 Import Files to SWORD




- Chart of Accounts
- Contacts File
- Customers
- Inventory Items

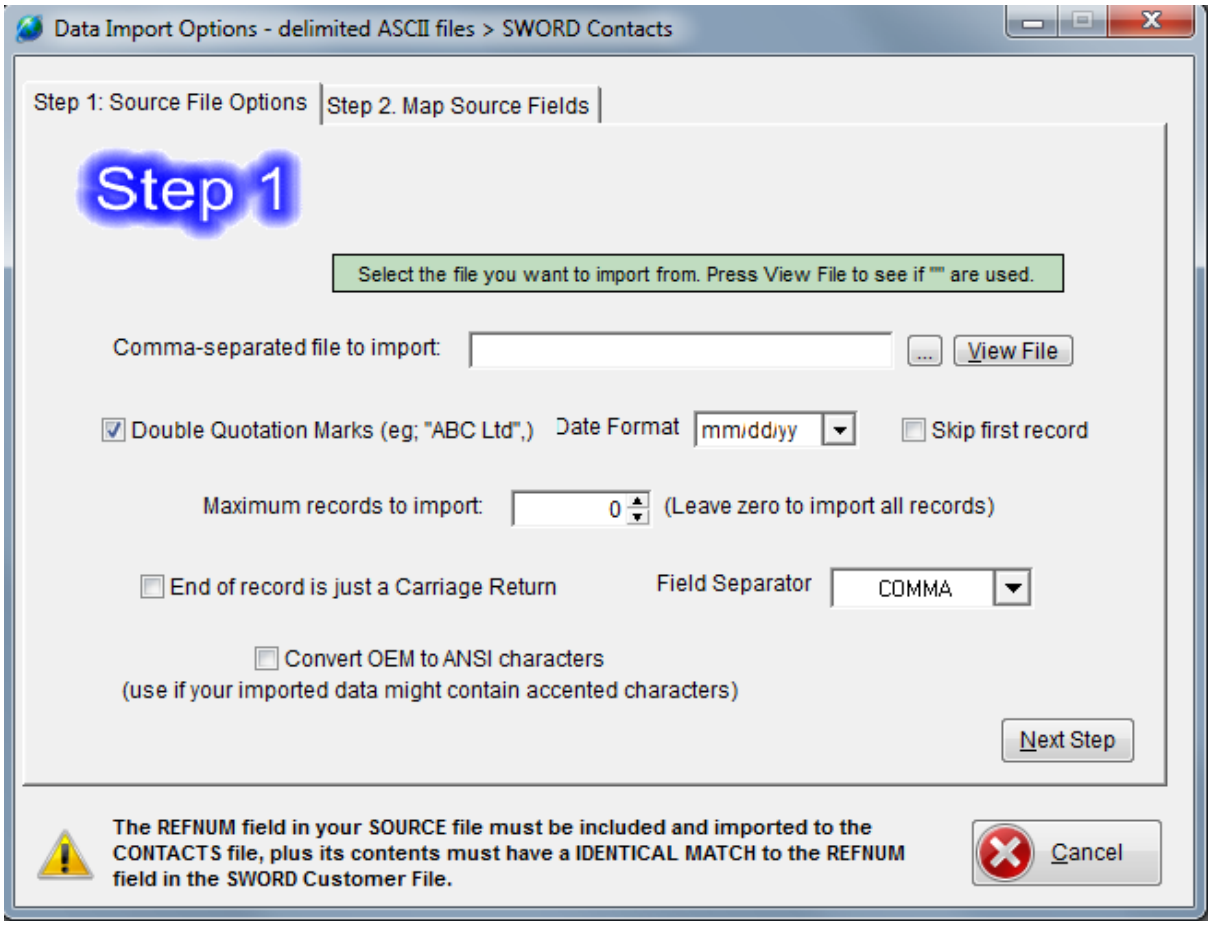
The above files may be IMPORTED from a CSV or EXCEL file type.

**User Tip:** Export the SWORD files using the Export SWORD Files option to create a template file to be re-imported with your data.




**Caution: Importing Files Requires EXCLUSIVE USE!**

**Step 1: Source File Options**




1. Navigate to a comma-separated file to import.
2. Optionally, you can view the file for verification purposes.
3. If COMMAS are used inside FIELDS then your data should have been prepared for DOUBLE QUOTATION MARKS.
4. Select how you wish to format fields with dates.

❑ Skip first record > Check this box if you wish to import only your data and not your header labels.

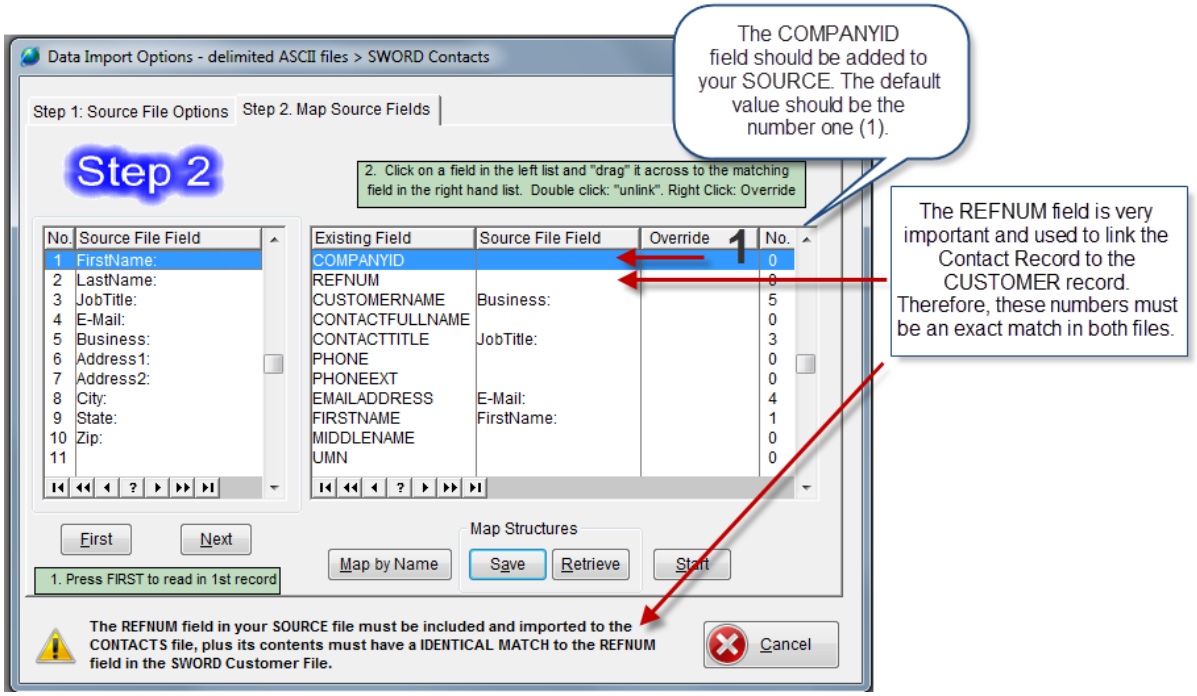
5. Maximum records to import may be set to a small number for testing your import. A ZERO will import ALL records.

❑ End of record is a Carriage Return > some databases require additional symbols to force a new line. You may need to experiment with this check box to verify that only a Carriage Return is needed.

6. Field Separator > Alternately, the file may be delimited with a TAB, SEMICOLON or PIPE symbol. You may need to experiment with another delimiter if a COMMA is not giving you the most desirable results.

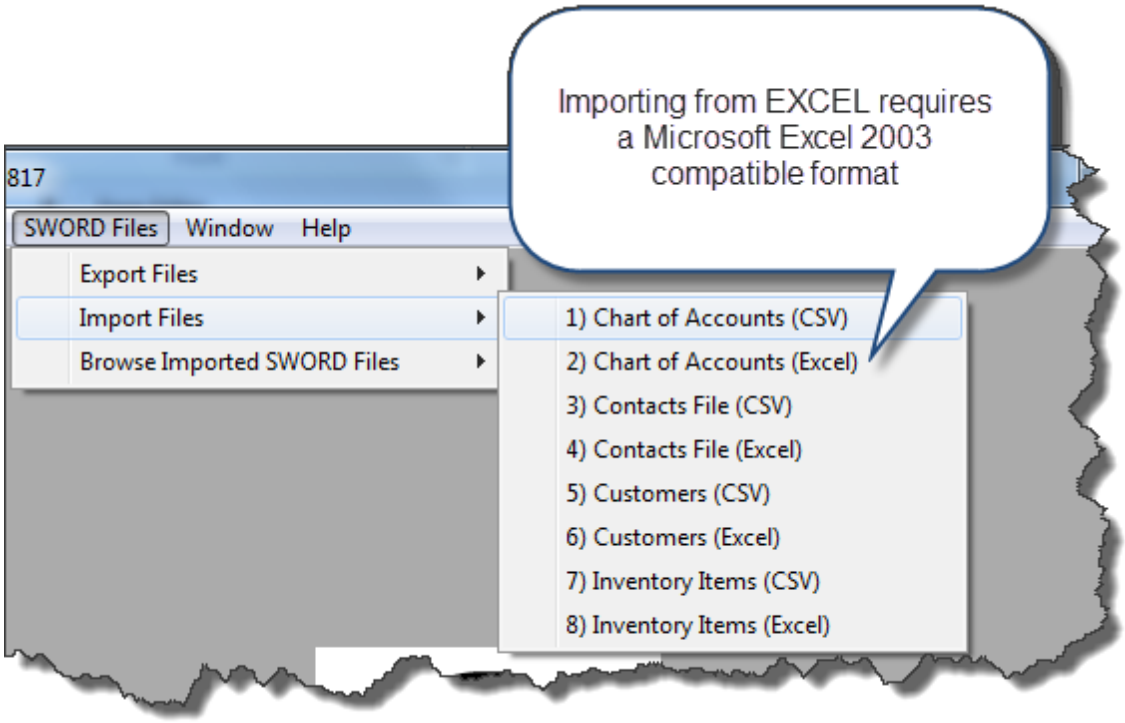
❑ Convert OEM to ANSI characters. (Use if your imported data might contain accented characters)

Click on the [**Next Step button**] to open the **Step 2. Map Source Fields** TAB.




1. Press the [**FIRST button**] to read in the 1st Record.
2. Click on a field in the left list (your SOURCE file) and "drag" it across to the matching field in the right hand list (SWORD fields).
3. Use the mouse "Double-click" in the right hand list (SWORD fields) to "unlink" the match.
4. Use the mouse "Right-Click" to enter a static value for the SWORD field. For example, the **COMPANYID field *must always contain the number 1*.**
5. The matching of fields may be saved as a "**Map Structure**." Once you have matched your fields to the SWORD fields then click on the [**Save button**].
6. The **Column labeled No**. in the **Source file** (Left-side panel) and **Existing File Field** (Right-side panel) indicates which column in the Source file shall be placed into the Existing field.
7. Use the [**Retrieve button**] to use a previously saved Map Structure.
8. Click on the [**Start button**] to begin the IMPORT.

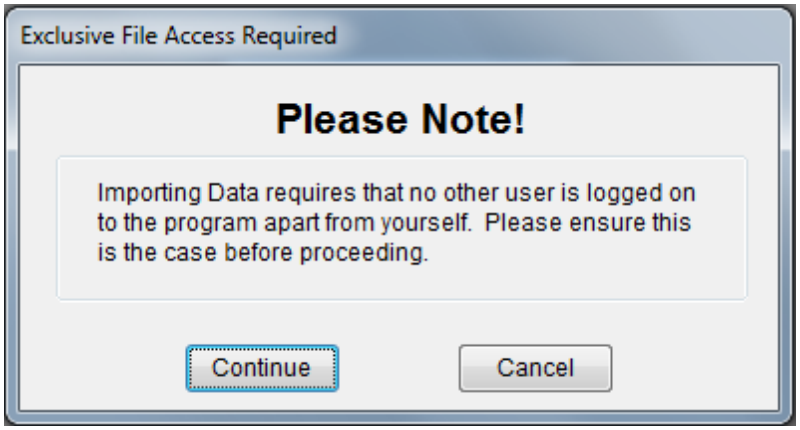
### 2.2.1 Import EXCEL to SWORD




- Chart of Accounts
- Contacts File
- Customers
- Inventory Items

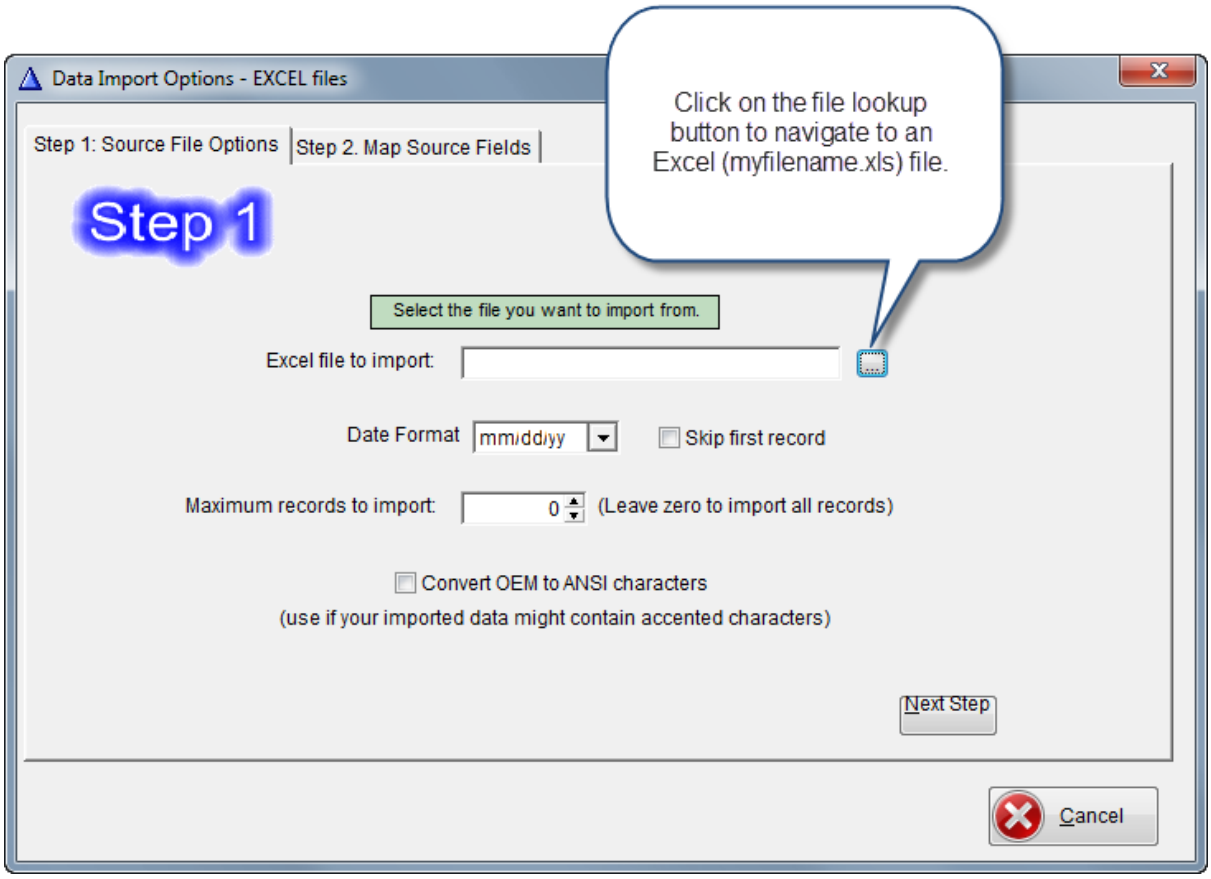
The above files may be IMPORTED from a MICROSOFT EXCEL 2003 compatible file type.

**User Tip:** Export the SWORD files using the Export SWORD Files option to create a template file to be re-imported with your data.



## Caution: Importing Files Requires EXCLUSIVE USE!

### Step 1: Source File Options



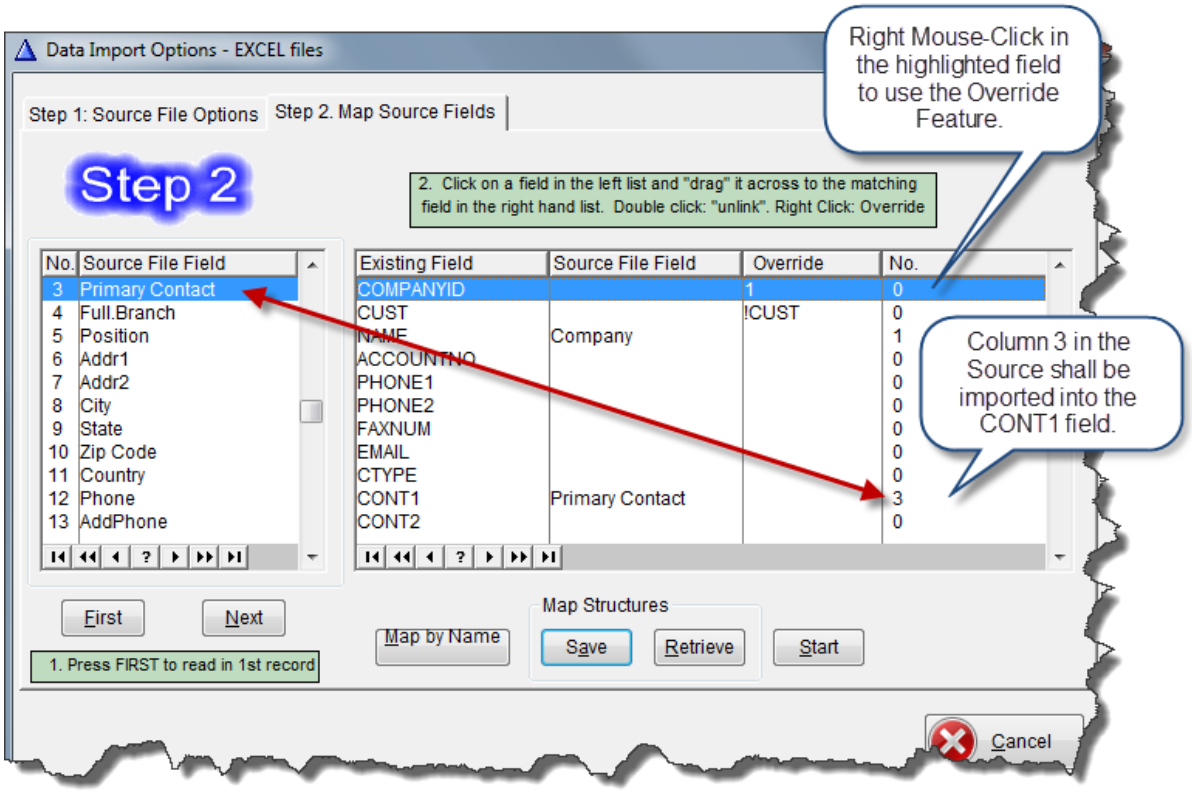

1. Navigate to an EXCEL.XLS file to import.
2. Select how you wish to format fields with dates.

❑ Skip first record > Check this box if you wish to import only your data and not your header labels.

3. Maximum records to import may be set to a small number for testing your import. A ZERO will import ALL records.

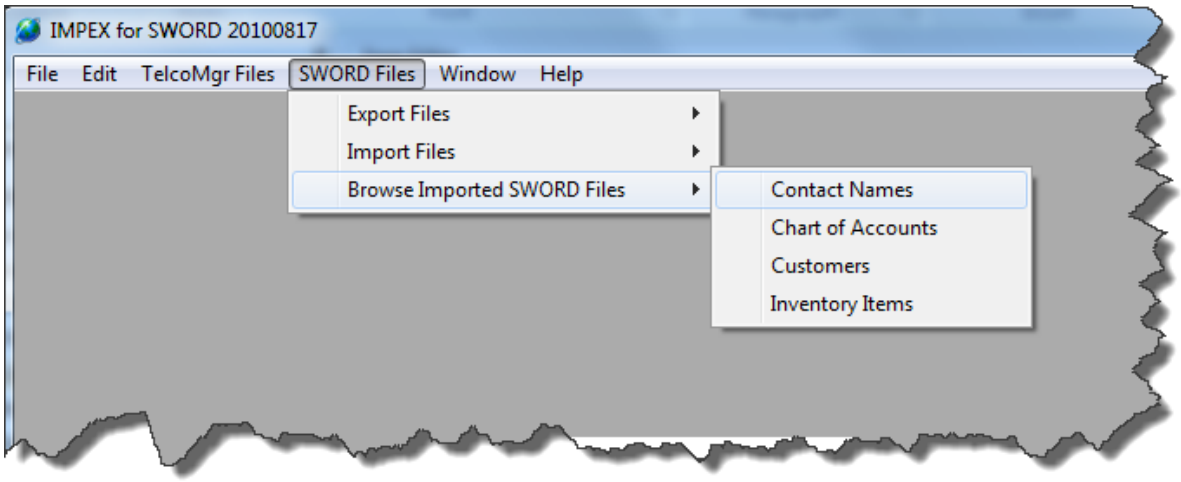
❑ Convert OEM to ANSI characters. (Use if your imported data might contain accented characters)

Click on the [**Next Step button**] to open the **Step 2. Map Source Fields** TAB.




1. Press the [**FIRST button**] to read in the 1st Record.
2. Click on a field in the left list (your SOURCE file) and "drag" it across to the matching field in the right hand list (SWORD fields).
3. Use the mouse "Double-click" in the right hand list (SWORD fields) to "unlink" the match.
4. Use the mouse "Right-Click" to enter a static value for the SWORD field. For example, the **COMPANYID field *must always contain the number 1*.**
5. The matching of fields may be saved as a "**Map Structure**." Once you have matched your fields to the SWORD fields then click on the [**Save button**].
6. The **Column labeled No**. in the **Source file** (Left-side panel) and **Existing File Field** (Right-side panel) indicates which column in the Source file shall be placed into the Existing field.
7. Use the [**Retrieve button**] to use a previously saved Map Structure.
8. Click on the [**Start button**] to begin the IMPORT.

## 2.3 Browse Imported Files to SWORD




For testing purposes, you may view your imported or existing data using the browse tables for the files listed below. IMPEX does not provide any editing capabilities. Imported files must be edited with SWORD.

- Contact Names
- Chart of Accounts
- Customers
- Inventory Items

# 3 TelcoMgr Import & Export Options

IMPEX provides the user the option of importing and exporting TelcoMgr data through several options, which includes CSV, Excel and HTML methods.

In order to get the most productivity from the IMPEX advanced feature, we recommend that you become familiar with this document and follow the embedded **USER TIPS**, which appear in various sections. After you have become familiar with the IMPEX capabilities try some practice scenarios, which will help you perfect your final IMPORT or EXPORT.

Related Topics:

TelcoMgr File Structures
Advanced Import Features
Advanced Export Features

Also, for exporting specific data based upon a query, a user can elect the Select Export

Records option from the main menu.

## 3.1 TelcoMgr File Structures

### Anatomy of a Database

This section briefly describes the fundamentals of database design. It is meant only to provide an overview of the subject for those who are not already thoroughly familiar with standard database design concepts and issues. Experienced users may want to move right on to the next chapter and skip this section.

**Definitions**
A database is a collection of information (data) in a system of tables, rows, and columns. The database is maintained by one or more computer programs or applications.

The basic unit of data storage is a column. A column is a storage place for information of a similar type. For example, one column might store a name and another column might hold a telephone number. A group of different columns that are logically related make up a row. A row contains all the information related to one subject. For example, all the columns containing information concerning one student (name, address, telephone number, student number, etc.) makes up one student's row. This would be similar to a file folder a school might keep for each student.

Collections of logically related rows make up a table. Using the same example, a collection of all students' rows makes up the student body table. This would be similar to the file cabinets where students' folders are kept.
Another way of looking at this is as a table or spreadsheet:

| Number | First Name | Last Name | Description |
|---|---|---|---|
| 768-90-3748 | Jim | Babbitt | Computer Science |
| 603-71-4468 | Randall | Babcock | Computer Science |
| 831-87-7376 | Brian | Babin | Computer Science |
| 980-10-7626 | James | DeSavia | Business |
| 251-05-8624 | Robert F | Bachman | Computer Science |
| 967-26-3524 | John H | Bagby | Law |
| 305-14-8528 | Debbie | Bagley | Business |
| 141-02-7461 | Alvin L | Bailey | Computer Science |
| 827-93-7660 | John | Bailey | Sociology |
| 188-67-7396 | Robin | Baley | Business |
| 286-95-9466 | Curtis | Bair | Law |
| 101-71-0630 | Jerry | Cade | Business |
| 661-47-3367 | Harry R | Cahoon | Law |
| 474-12-9470 | David R | Cain | Computer Science |
| 947-64-8335 | David | Calabrese | Sociology |

In this format, the entire table is a file cabinet, with rows (folders) and columns (data about one row).
A database is a collection of related tables. This is similar to a bank of file cabinets where the entire school records are kept. One file cabinet might hold the files with

students' personal data, another with class enrollment information, and another with faculty information.

A relational database is a collection of tables with defined relationships between them. Effective database design breaks the data into related tables that are joined together through linking columns. This will be covered in detail later in this section.

**Summary**

- One or more columns combine to form a row.
- One or more rows combine to form a table.
- A collection of related tables is a database.*This documentation reflects the files structures for TelcoMgr version 5 and subject to change without notice.

**User Tip:** Export a TelcoMgr Locaton, Circuits and/or Directory file as a **Comma Separated Value** (**CSV**) file to create a **file template** for importing data.

**User Tip:** IMPORT THE LOCATION FILE FIRST SO TELCOMGR CAN CREATE THE CUSTOMER SYSTEM ID (CusSysID) number automatically. The appropriate CusSysID must appear in each record that is imported into the CIRCUITS and DIRECTORY files.

CSV or EXCEL data may be imported into the following TelcoMgr files.

1) telco.tps - The Customer/Location file is classified as the PARENT file and the master relationship is set by the CusSysID field, which is common to the CHILD files below:
   a) circuits.tps - The CIRCUITS file contains the physical and logical data descriptions of each circuit, including lines, features, services, charges and taxes. The CusSysID field is related to the telco.tps CusSysID field.
   b) directry.tps - The DIRECTORY file contains users names, extension numbers and DID numbers and is related to the telco.tps file.
   c) cpe.tps - The CPE file may be used for variety of purposes, which includes components, systems, individual equipment, etc. and the PARENT to the COMPONENTS file.
      i) componts.tps - The COMPONENTS file is a CHILD of the CPE file and includes detailed information about equipment/system components.

**Reserved Fields**

The TelcoMgr [Reserved] fields are numeric auto-increment fields which are automatically generated by TelcoMgr, therefore data should not be imported into a [Reserved] fields.

Review the field layout for each file to learn which fields are **[Reserved] fields**.

telco.tps
circuits.tps

directry.tps
cpe.tps

### 3.1.1 Data Types

Columns can store many different types of data, but each individual column may hold only one type. When a column is defined, its data type is specified. This enables it to efficiently store that type of data. For example, to store a number from 0 to 100, using a column defined as a single BYTE takes less space than one defined as a decimal number column (a byte can hold an unsigned whole number between 0 and 255).

#### Data Types Used in TelcoMgr

**Byte** : A numeric constant. If omitted, the initial value is zero, unless the AUTO attribute is present.
**Date** : DATE declares a four-byte date variable. A DATE used in a numeric expression is converted to the number of days elapsed since December 28, 1800. The valid Standard Date range is January 1, 1801 through December 31, 9999. Using an out-of-range date produces unpredictable results.
**Decimal** : A required numeric constant containing the total number of decimal digits (integer and fractional portion combined) in the variable. The maximum length is 31. Example = 10,000.50
**Long** : LONG declares a four-byte signed integer, using the Intel 8086 long integer format. The high-order bit is the sign bit (0 = positive, 1 = negative). Negative values are represented in standard two's complement notation.
**Memo** : Memo fields are large String fields used for alpha and/or numeric character or binary data elements.
**String** : STRING declares a fixed-length alpha and/or numeric character string. The memory assigned to the STRING is initialized to all blanks unless the AUTO attribute is present.

#### Format Examples and Pictures:

STRING(40) indicates a STRING field with 40 characters;
DECIMAL(9,2) indicates a DECIMAL field with 9 characters, 2 are to the right of the decimal, with attributes (minus, comma, period, currency symbol); 99,900.99;
LONG(-14) indicates a numeric field with 14 characters with a minus attribute.

### 3.1.2 telco.tps file

| Field Name | Attributes |
|---|---|
| CompanyID | LONG |
| CusSysID | LONG |
| [RESERVED] | |
| REFNUM | LONG |
| NAME | STRING(72) |
| ContactForTelco | STRING(41) |
| ContactJobTitle | STRING(40) |

**Developer Comments**
The CompanyID should be the numeric 1. Do not import this field TelcoMgr will generate the CusSysID
Quickbooks Reference #

| | |
|---|---|
| ContactPhone | STRING(15) |
| ContactFax | STRING(15) |
| ContactEmail | STRING(41) |
| EADDR1 | STRING(41) |
| EADDR2 | STRING(41) |
| EADDR3 | STRING(41) |
| EADDR4 | STRING(41) |
| EADDR5 | STRING(41) |
| SADDR1 | STRING(41) |
| SADDR2 | STRING(41) |
| SADDR3 | STRING(41) |
| SADDR4 | STRING(41) |
| SADDR5 | STRING(41) |
| LECName | STRING(40) |
| LECAcctCode | STRING(20) |
| LECBillToName | STRING(40) |
| IECName | STRING(40) |
| IECAcctCode | STRING(20) |
| IECBillToName | STRING(40) |
| LDPICCode | STRING(20) |
| EditDate | LONG |
| EditTime | LONG |
| TenXXXDialing | STRING(3) |
| GroupAccountCode | STRING(20) |
| LECPICCode | STRING(20) |
| AlternateContact | STRING(41) |
| AltContactPhone | STRING(15) |
| AltContactEmail | STRING(41) |
| UserDefinedOrder | STRING(40) |
| OECName | STRING(40) |
| OECAcctCode | STRING(20) |
| OPICCode | STRING(20) |
| MSContract | STRING(40) |
| MSContractStart | LONG |
| MSContractTerm | DECIMAL(3,0) |
| MSContractEnd | LONG |
| MSMonthlyFee | DECIMAL(9,2) |
| MSComments | STRING(1000) |
| CalcMSCEnd | BYTE |
| OnMSC | BYTE |
| MSCSignDate | LONG |
| MSCReminder | LONG |
| AssignedToResource | STRING(60) |
| AttachImage | STRING(255) |
| AtachPdf | STRING(255) |
| HtmlEmail | STRING(255) |

### 3.1.3 circuits.tps file

| **Field Name and Attributes** | | Developer Comments |
|---|---|---|
| CusSysID | LONG | The CusSysID should match the Customer/ Location telco.tps cussysid. |
| LECName | STRING(40) | |
| LECAcctCode | STRING(20) | |
| IECName | STRING(40) | |

```
IECAcctCode             STRING(20)
ExistingNew             STRING(8)
BTN                     STRING(3)
MTN                     STRING(3)
ClassOfService          STRING(20)
Line                    STRING(12)
PIC                     STRING(20)
PTN                     STRING(12)
LNP                     STRING(3)
HuntGroup               STRING(20)
ExistApp                STRING(20)
PostCutAction           STRING(20)
NewApp                  STRING(20)
PCAEffectiveDate        LONG
PCAEffectiveTime        LONG
EditDate                LONG
EditTime                LONG
Paths                   DECIMAL(3,0)
CircuitNo               STRING(40)
ServOrdNo               STRING(20)
ServOrdDate             LONG
MonthlyAmount           DECIMAL(7,2)
ServOrdInstallDate      LONG
DNIS                    STRING(20)
TFN                     STRING(3)
ExchangeServices        STRING(20)
LFS                     STRING(1)
TenXXXDialing           STRING(3)
BindingPostID           STRING(20)
Qty                     DECIMAL(7,0)
SeqInHunt               DECIMAL(7,0)
CallForwardTo           STRING(12)
ReferenceNotes          STRING(1000)
LECPIC                  STRING(20)
SFLTagField             STRING(20)
SysID                   LONG
[RESERVED]
PCAStatus               STRING(20)
PCAStatusDate           LONG
OECName                 STRING(40)
OECPIC                  STRING(20)
OECAcctCode             STRING(20)
DLCI                    STRING(15)
PortSpeed               STRING(10)
PVC                     STRING(15)
OnTermContract          BYTE
SignDate                LONG
BeginDate               LONG
TermMonths              DECIMAL(3,0)
RenewalDate             LONG
ReminderDate            LONG
PrimaryContractNo       STRING(40)
SecondaryContractNo     STRING(40)
```

Do not import data into **[RESERVED]** fields.

| | |
|---|---|
| AdendumNo | STRING(40) |
| CalcEnd | BYTE |
| PDFattachment | STRING(255) |
| ImageAttachment | STRING(255) |
| HTMLattachment | STRING(255) |

### 3.1.4 directry.tps file

| Field Name | Attributes |
|---|---|
| CusSysID | LONG |
| Salutation | STRING(10) |
| FirstName | STRING(20) |
| MiddleInitial | STRING(2) |
| LastName | STRING(20) |
| FullName | STRING(50) |
| Nickname | STRING(20) |
| JobTitle | STRING(40) |
| PhoneExt | DECIMAL(7,0) |
| VMExt | DECIMAL(7,0) |
| EmailAdd | STRING(30) |
| LocationOfEquipment | STRING(40) |
| Department | STRING(20) |
| Line1Desc | STRING(20) |
| Line1 | STRING(12) |
| Line2Desc | STRING(20) |
| Line2 | STRING(12) |
| StationEquipment | STRING(40) |
| DirectoryOptions | STRING(4) |
| RmNameNumber | STRING(20) |
| SysID | LONG |
| [RESERVED] | |

The CusSysID must match the appropriate Telco.tps CusSysID

### 3.1.5 cpe.tps file

| Field Name | Attributes |
|---|---|
| CompanyID | LONG |
| RefNum | LONG |
| [Reserved] | |
| AccountNo | STRING(41) |
| Name | STRING(72) |
| LocationOfEquip | STRING(40) |
| Vendor | STRING(40) |
| Manufacturer | STRING(40) |
| TypeOfCPE | STRING(40) |
| ModelID | STRING(40) |
| Description | STRING(80) |
| RoomNumber | STRING(20) |
| SoftwareRelease | STRING(20) |
| SoftwareDate | LONG |
| DateInstalled | LONG |
| AcquisitionType | STRING(20) |
| WarrantyExpiry | LONG |
| MaintenancePlanNo | STRING(20) |
| MaintenanceExpiry | LONG |

The CompanyID should be the numeric 1. The CusSysID should match the Customer/ Location telco.tps cussysid.

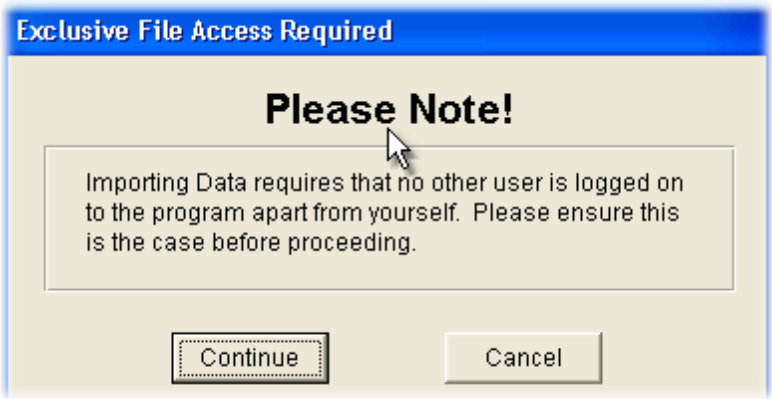
| | |
|---|---|
| ConfigurationDetail | STRING(256) |
| BackupPowerDevice | STRING(20) |
| RemoteAccessNumber | STRING(20) |
| RemoteAccessExt | STRING(20) |
| RemoteAccessLogin | STRING(20) |
| RemoteAccessPasscode | STRING(20) |
| CusSysID | LONG |
| CPEID | LONG |
| IPAddress | STRING(255) |
| AgreementNo | LONG |
| PrimaryMSC | BYTE |
| OnMSC | BYTE |
| MSContractStart | LONG |
| MSContractTerm | DECIMAL(3,0) |
| MSAnnualFee | DECIMAL(9,2) |
| CalcMSCEnd | BYTE |
| AssignedToResource | STRING(60) |
| NameOfPlan | STRING(40) |
| PaymentOption | STRING(20) |
| OptionPayment | DECIMAL(9,2) |

#### 3.1.5.1 CPE Components (componts.tps) file

| Field Name | Attributes |
|---|---|
| CompanyID | LONG |
| Refnum | LONG |
| TypeOfCPE | STRING(40) |
| Quantity | LONG |
| Manufacturer | STRING(40) |
| PartNumber | STRING(20) |
| Description | STRING(40) |
| CusSysID | LONG |
| CMPID | LONG |
| [Reserved] | |
| SerialNoID | STRING(40) |
| Location | STRING(20) |
| TypeOfPorts | STRING(20) |
| QtyofPorts | DECIMAL(3,0) |
| Comments | STRING(255) |
| ExtQtyofPorts | DECIMAL(9,0) |
| SysID | LONG |
| WarrantyExpiry | LONG |
| MaintenanceExpiry | LONG |

The CompanyID should be the numeric 1. Refnum is a Quickbooks and/or SWORD Reference Number.
The CusSysID should match the Customer/ Location telco.tps cussysid.

## 3.2 Advanced Import




**Single User**: Importing Data requires that no other user is logged on to TelcoMgr apart from a single TelcoMgr Uitility program user.

**Import Features:**

**Import** from **Comma Separate Value** (**CSV**) or **Excel** standard files.

**Duplicate** checking can be enabled at the end-user levels. The user has the option to process, ignore duplicates or update the original record. The contents of the dupes can be saved to a text file. At the end of the import run the Duplicates text file can be viewed/printed.

File information such as field names,types,sizes is read straight from the database header record and displayed in a browse.

The user can then match up fields by a **Map by Name** button (for similar labels) and then drag and drop. These Map Structures can be saved for future use.

At runtime the user can "override" incoming text for a particular field with their own choice of text (or date).

For testing purposes a set number of records can be imported.

Records are added to your existing data file and if there is an Auto Number key the autonumber field will be incremented automatically.

For ASCII or CSV files there is the option to skip the header record (useful if this record is just field names) and allow double quote marks, or not.

**Files allowed for Import**:

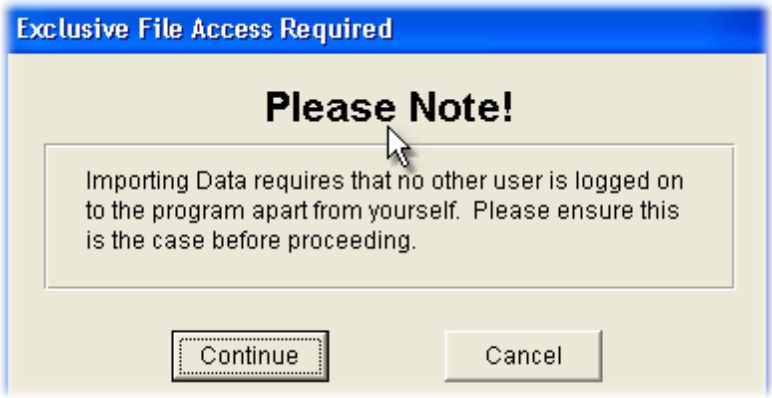

- Circuits
- Customer/Location
- Directory

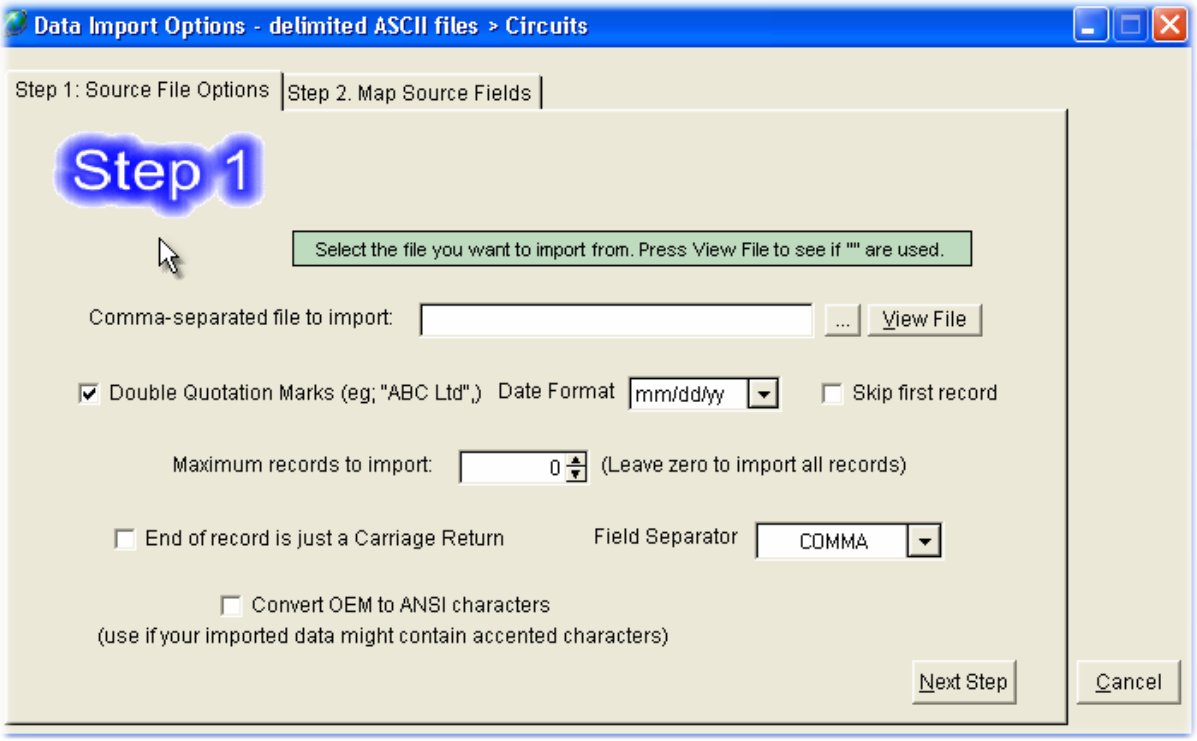
**User Tip:** Export a TelcoMgr Locaton, Circuits and/or Directory file as a **Comma Separated Value** (**CSV**) file to create a file template for importing data.

**User Tip:** IMPORT THE LOCATION FILE FIRST SO TELCOMGR CAN CREATE THE CUSTOMER SYSTEM ID (CusSysID) number automatically. The appropriate CusSysID must appear in each record that is imported into the CIRCUITS and DIRECTORY files.

**User Tip:** Do not import data into `[RESERVED]fields`.

### 3.2.1 Importing CSV Files

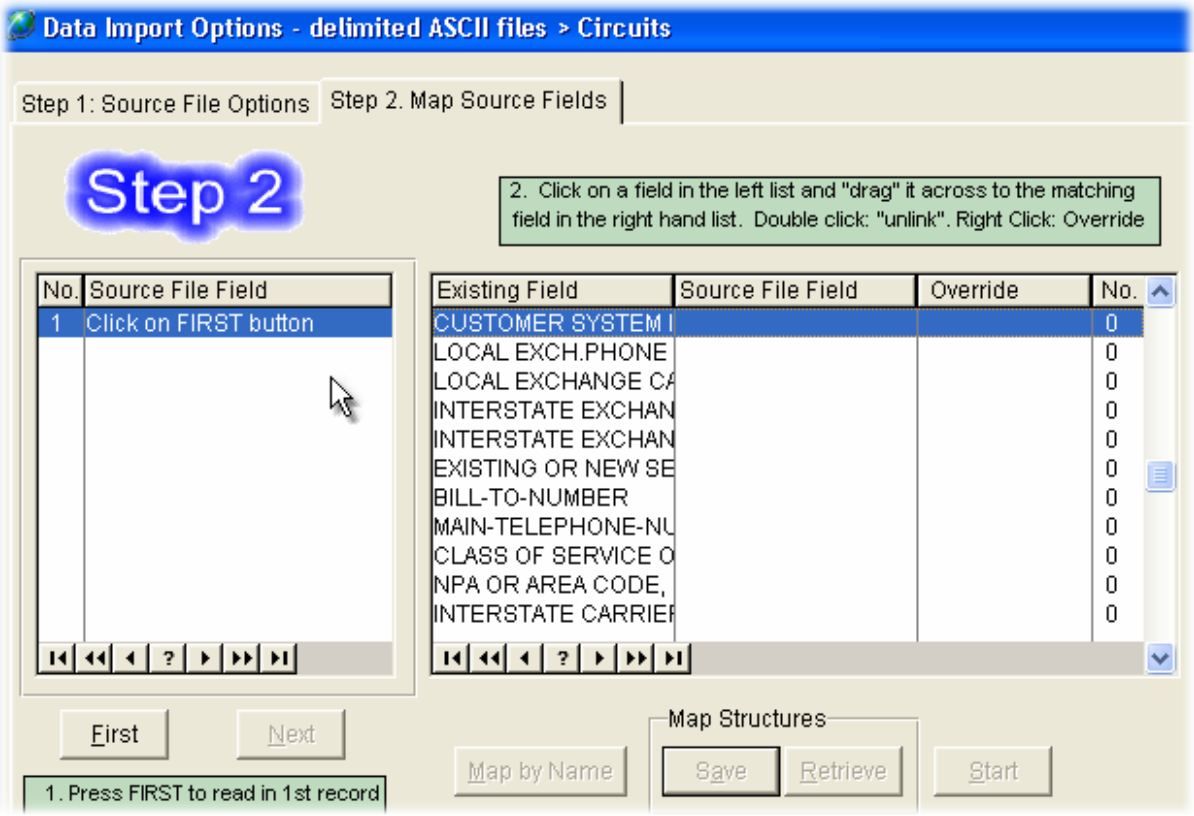





**Step 1 Notes:**

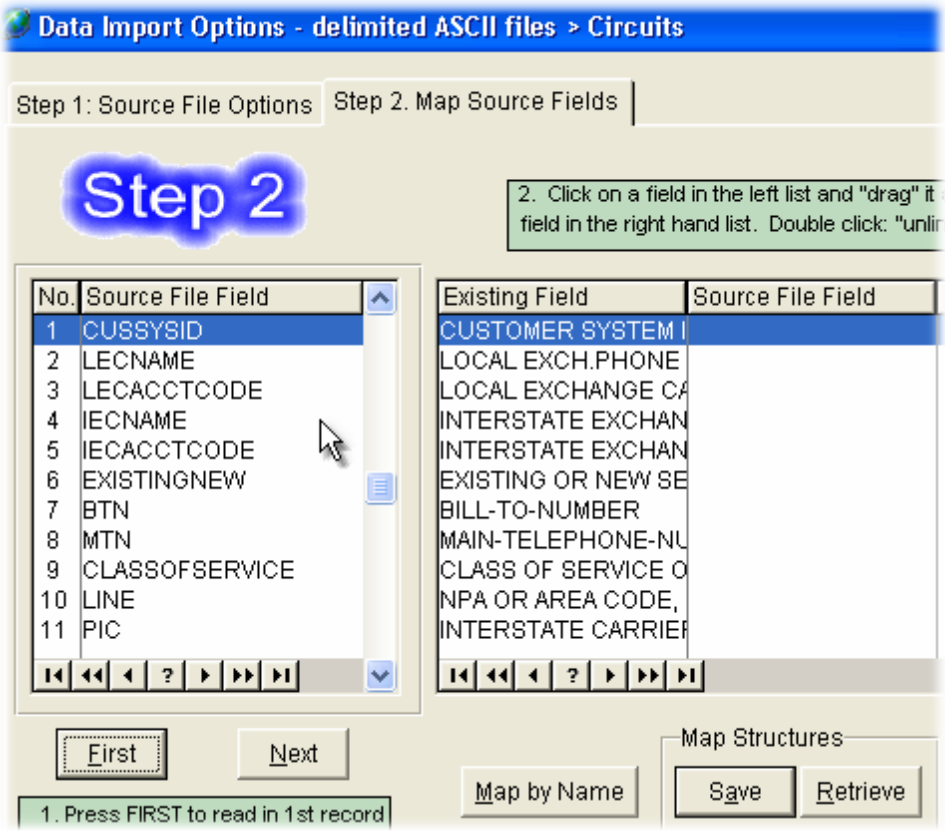
[Comma-separated file to import] use the [...] **elipse button** to navigate and select your file.

Use the [**View File**] button to verify the file that you wish to import.

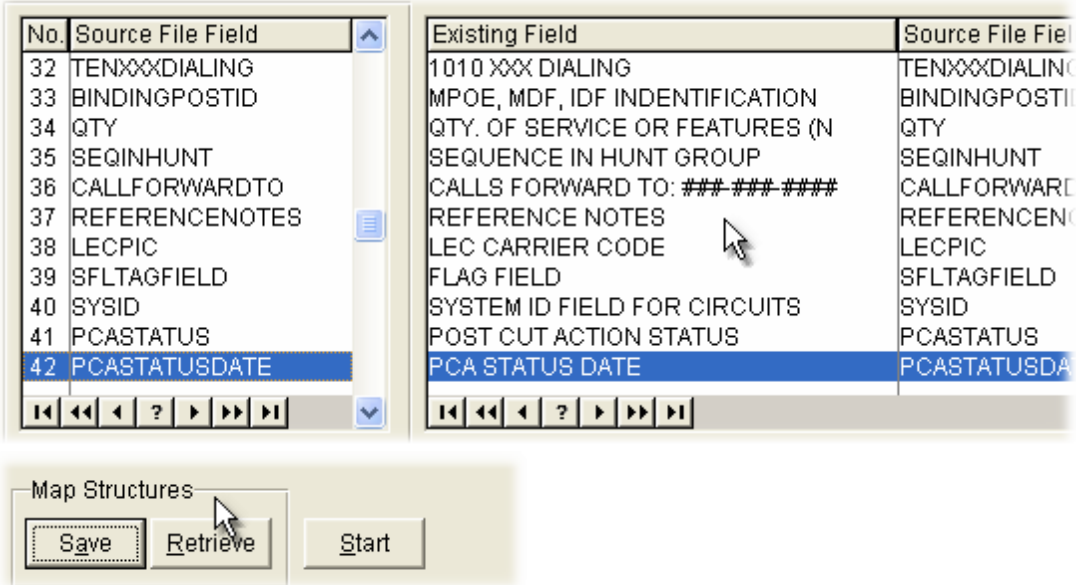



**Step 2 Notes:**

The small browse table on the left is where your import file will apppear. Click on the **[First] button** to load your data into the browse table. The browse table on the right shows the TelcoMgr file descriptions.




To **Map** the import fields to the TelcoMgr file, click on a field from the left browse table over to the appropriate line on the right browse table.




The **Map** of the import fields can be **Saved** and **Retrieved** for reuse.

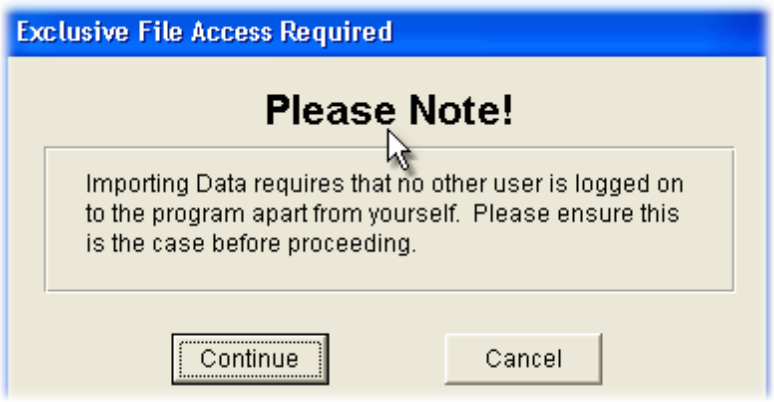
To complete the **Import Process** click on the **[Start] button.**

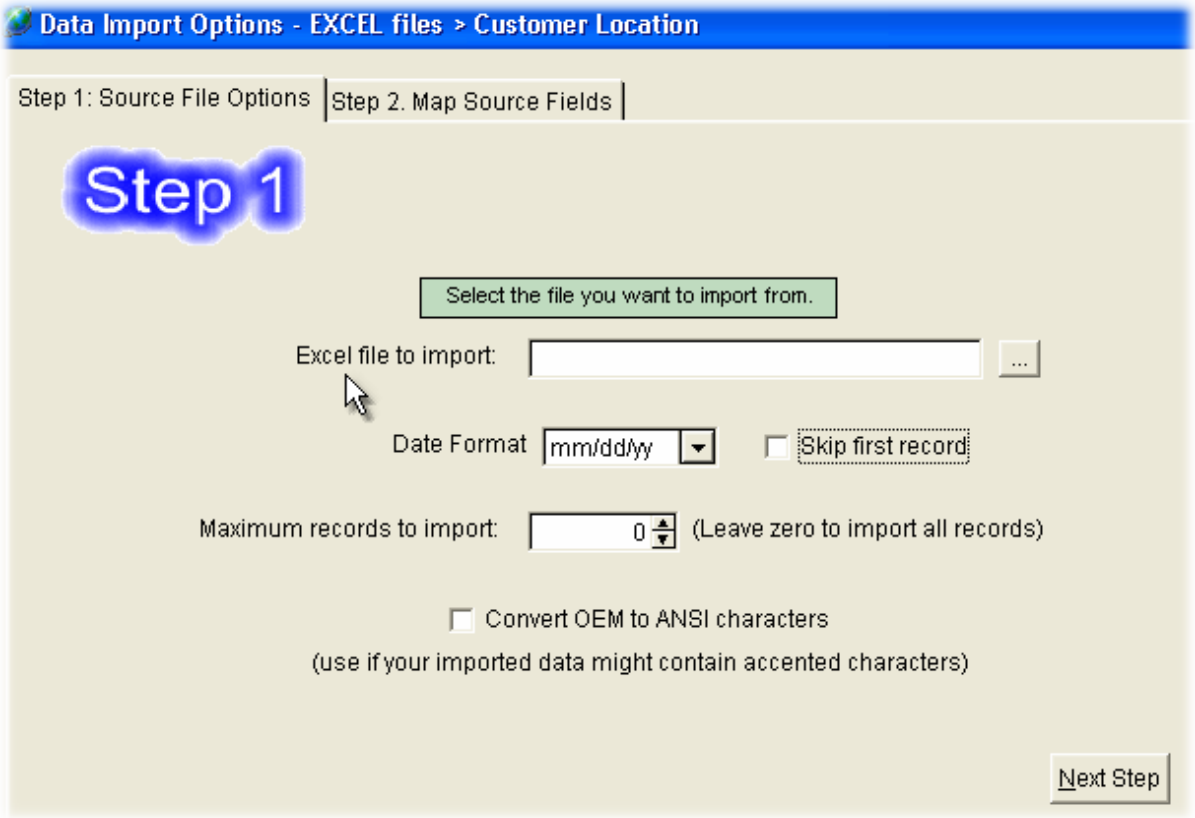
**User Tip:** Export a TelcoMgr Locaton, Circuits and/or Directory file as a **Comma Separated Value** (**CSV**) file to create a file template for importing data.

**User Tip:** IMPORT THE LOCATION FILE FIRST SO TELCOMGR CAN CREATE THE CUSTOMER SYSTEM ID (CusSysID) number automatically. The appropriate CusSysID must appear in each record that is imported into the CIRCUITS and DIRECTORY files.

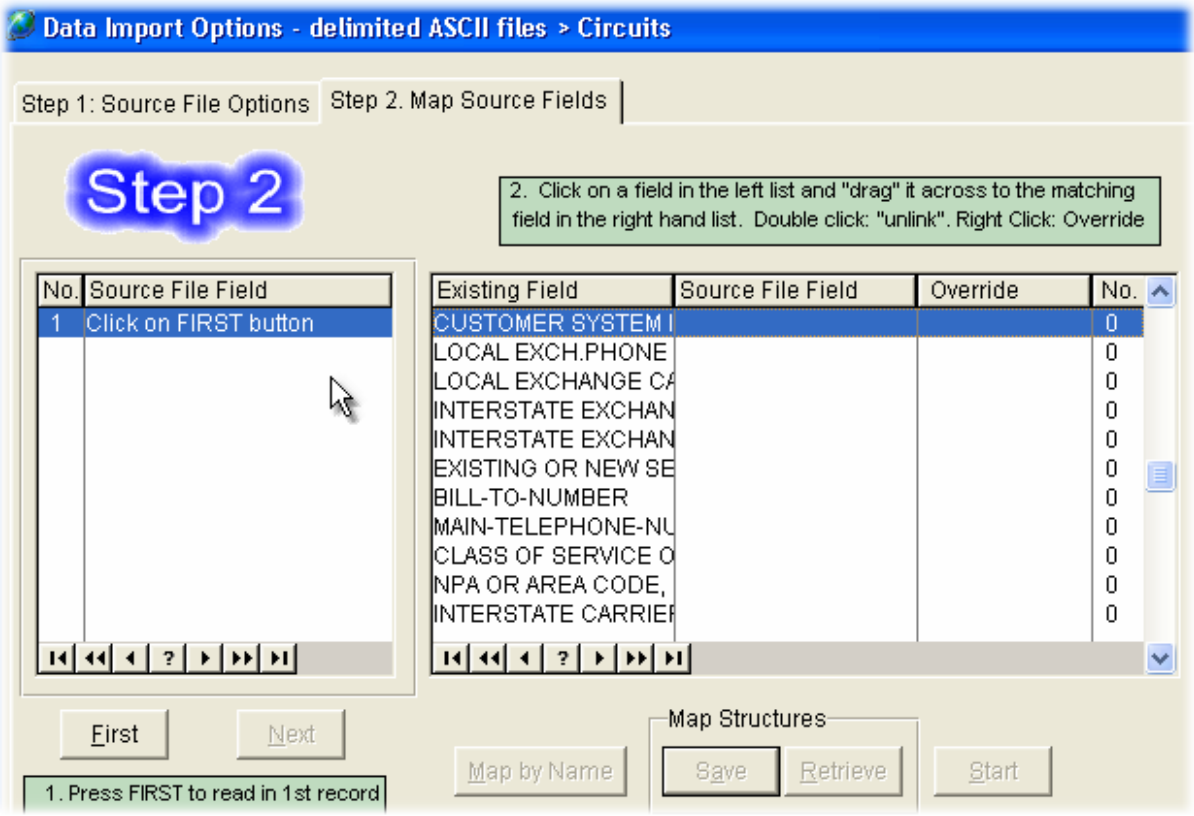
**User Tip:** Do not import data into `[RESERVED]fields`.

### 3.2.2 Importing Excel Files



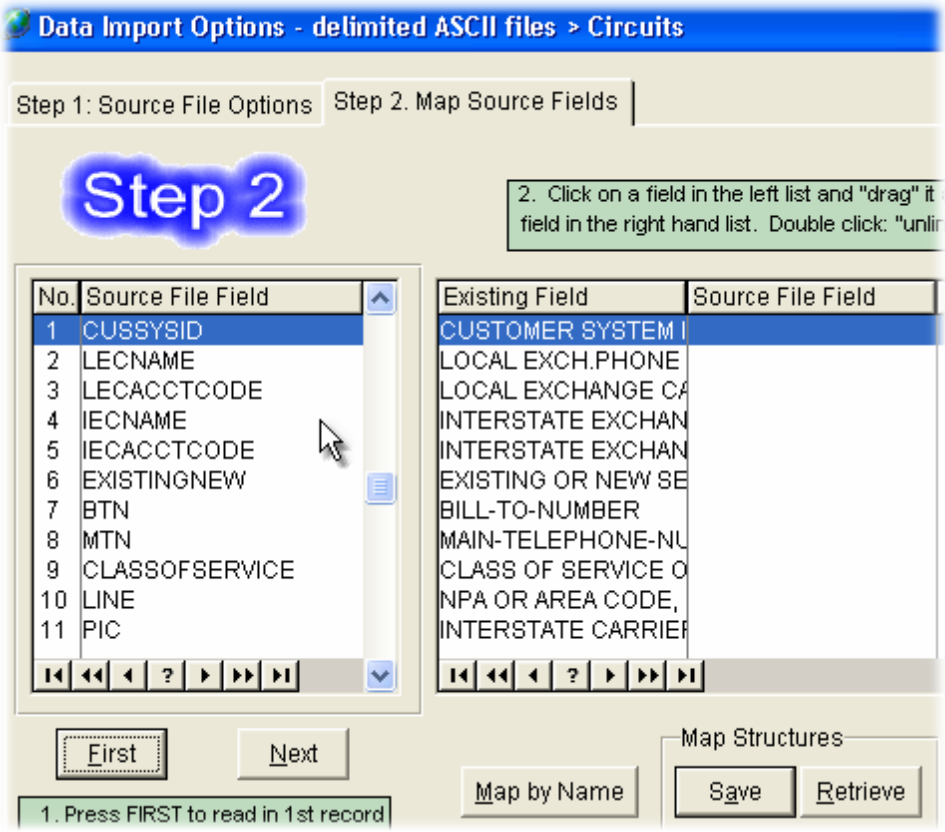



**Step 1 Notes:**
[**Excel File to Import**] use the [...] **elipse button** to navigate and select your file.
Use the [**View File**] button to verify the file that you wish to import.

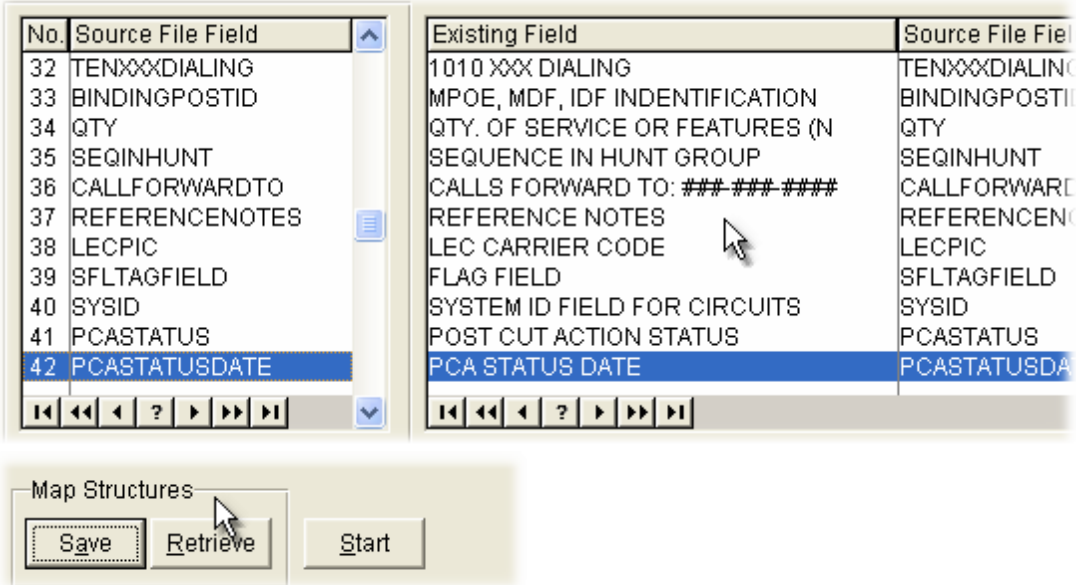



**Step 2 Notes:**

The small browse table on the left is where your import file will apppear. Click on the **[First] button** to load your data into the browse table. The browse table on the right shows the TelcoMgr file descriptions.




To **Map** the import fields to the TelcoMgr file, click on a field from the left browse table over to the appropriate line on the right browse table.




The **Map** of the import fields can be **Saved** and **Retrieved** for reuse.

To complete the **Import Process** click on the **[Start] button.**

**User Tip:** Export a TelcoMgr Locaton, Circuits and/or Directory file as a **Comma Separated Value** (**CSV**) file to create a file template for importing data.

**User Tip:** IMPORT THE LOCATION FILE FIRST SO TELCOMGR CAN CREATE THE CUSTOMER SYSTEM ID (CusSysID) number automatically. The appropriate CusSysID must appear in each record that is imported into the CIRCUITS and DIRECTORY files.

**User Tip:** Do not import data into `[RESERVED]fields`.

## 3.3 Advanced Export


**Export Features:**

The above listed files can be exported from TelcoMgr as **CSV**, **HTML** or **Excel** files.

**HTML options:** end user control over page and table attributes (font,color,width, cellspacing etc.), inclusion of user header/footer HTML (for example standard navigation bars) so the look and feel of your website can be maintained, table fields can be given a URL link, data can be output to one page or multiple pages. In the latter case Next/Previous links and page numbers are placed on each page generated.

The user can define which fields are included in the export file. So for example, the user could exclude confidential data fields from being exported. These Selections can be saved for future use.

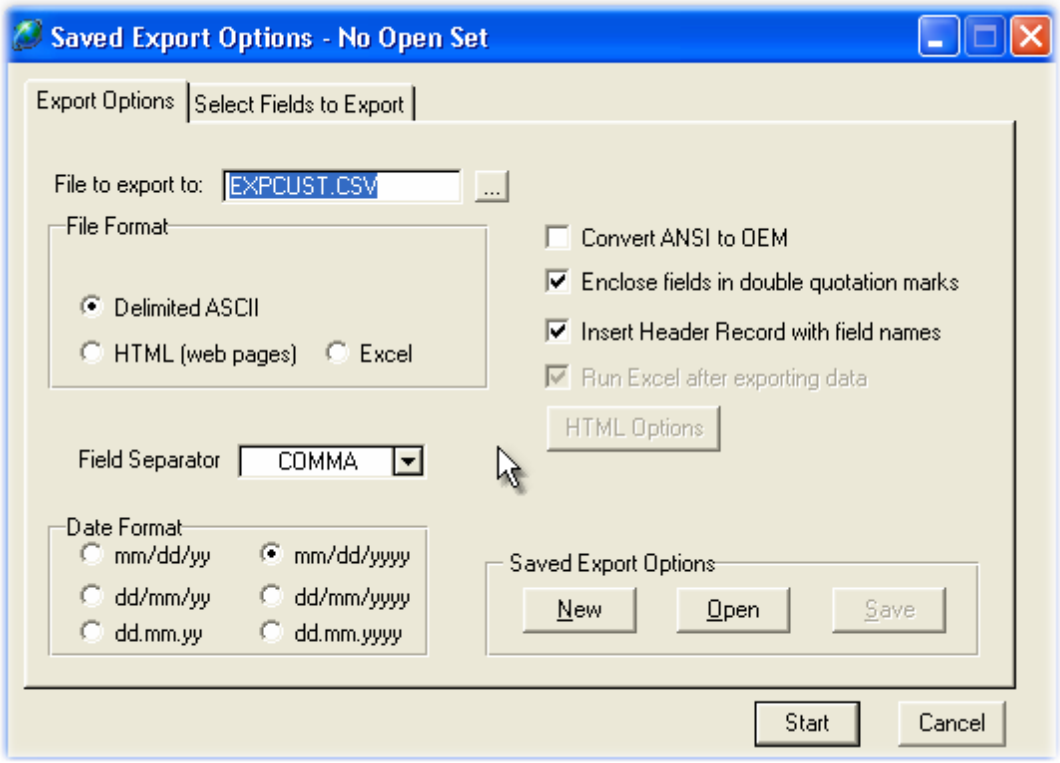
For **ASCII** files there are options for header record, double quotes and date format. Date fields are automatically detected by the templates.

A new field can be added at export time - so for example a field containing First Name + Middle Name + Last Name could be composed and added to the export file.

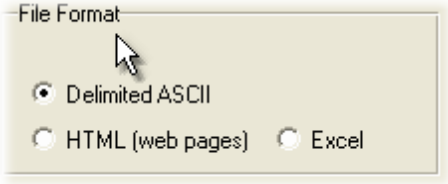
**USER TIP:** Export all data and you will see the information that you are looking for and the field

column where it appears. Then go back into IMPEX and Search on the appropriate Column name or **field description**. If you are using the Search and Tag feature in the field drop down selection box shows the Field Descriptions, which are shown to the user when the mouse cursor lands on or hovers over an entry field in TelcoMgr.

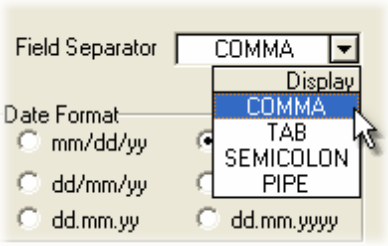
### 3.3.1 Exporting as CSV Files




**[File to export to] field**, use the **[...] elipse button** to navigate and select the file that you wish to import.

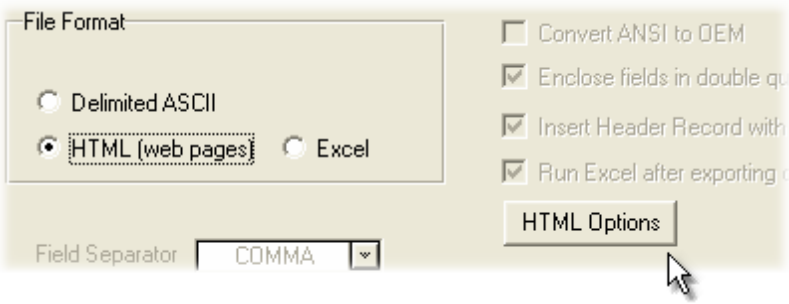



Select a **file format** that you wish to import.



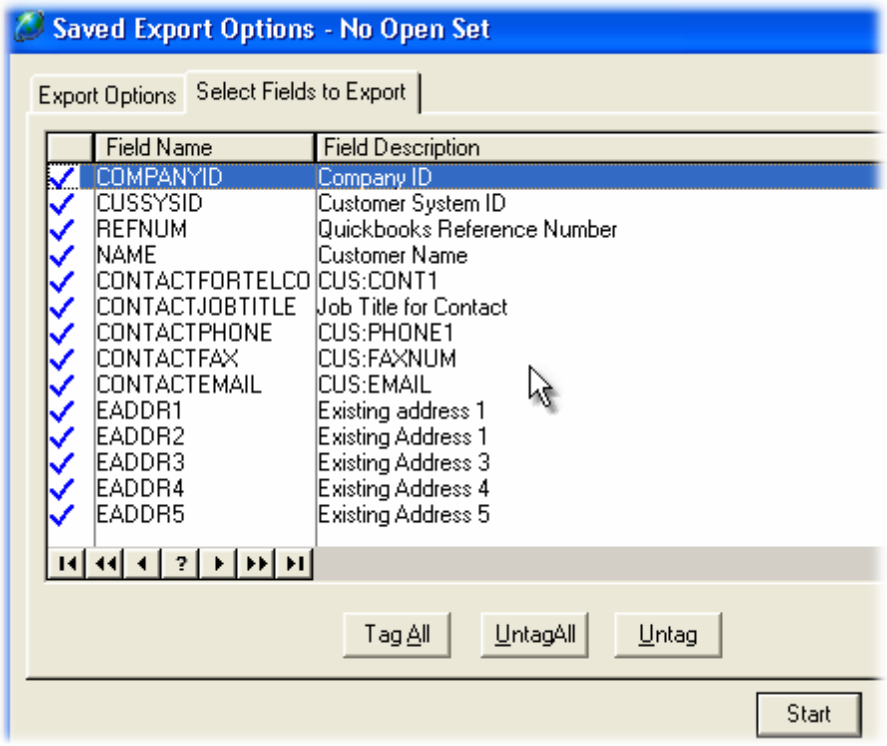

Select from one of four **Field Separator**(s):

- Comma
- Tab
- Semicolon
- Pipe



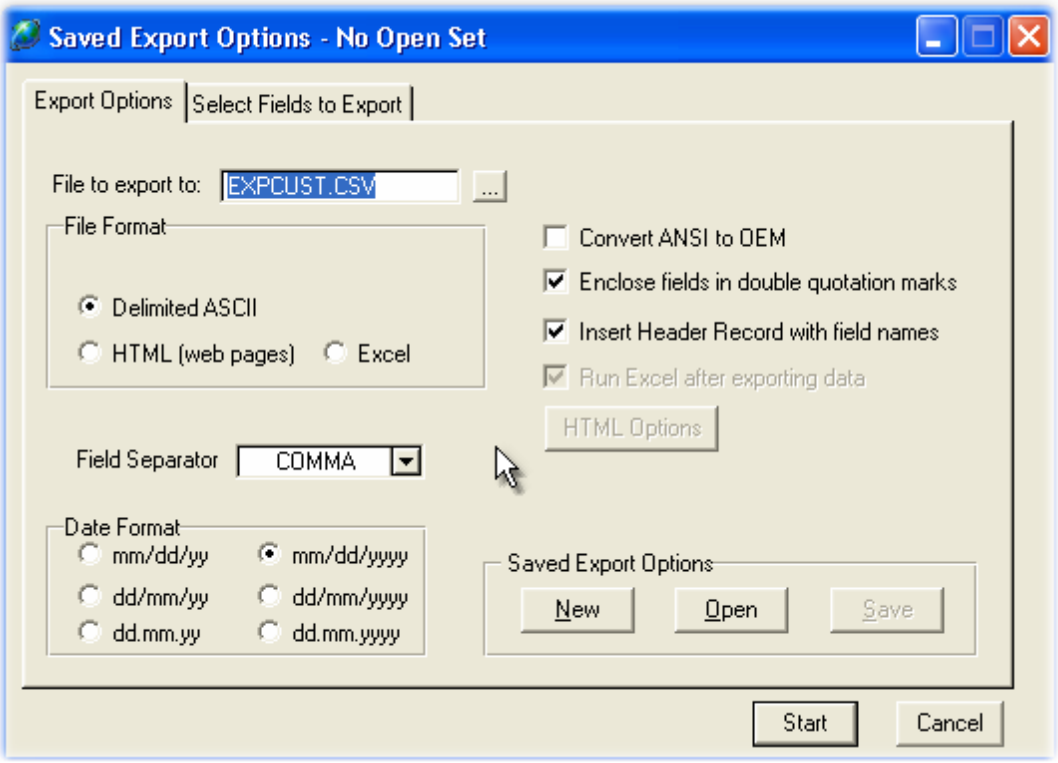

The HTML Option **radio button** will activate the **[HTML Options] button**.

The **Excel** radio button will activate the Header and **Run Excel option**s.

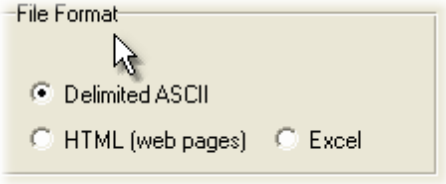



Fields may be **tagged** for export.

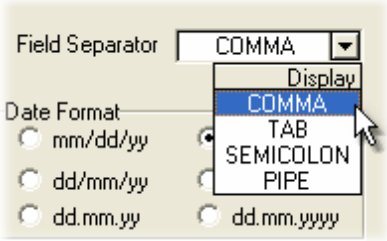
### 3.3.2 Exporting as HTML Files




**[File to export to] field**, use the **[...] elipse button** to navigate and select the file that you wish to import.



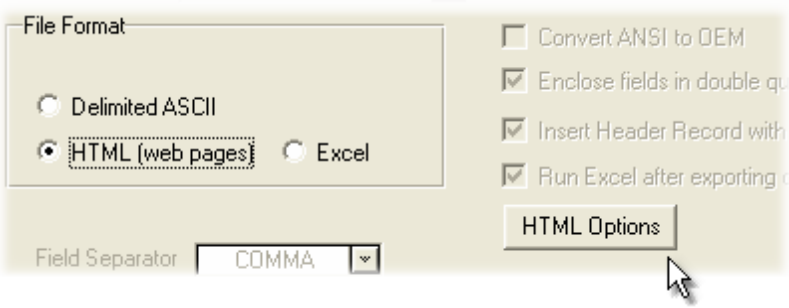

Select a **file format** that you wish to import.




Select from one of four **Field Separator**(s):

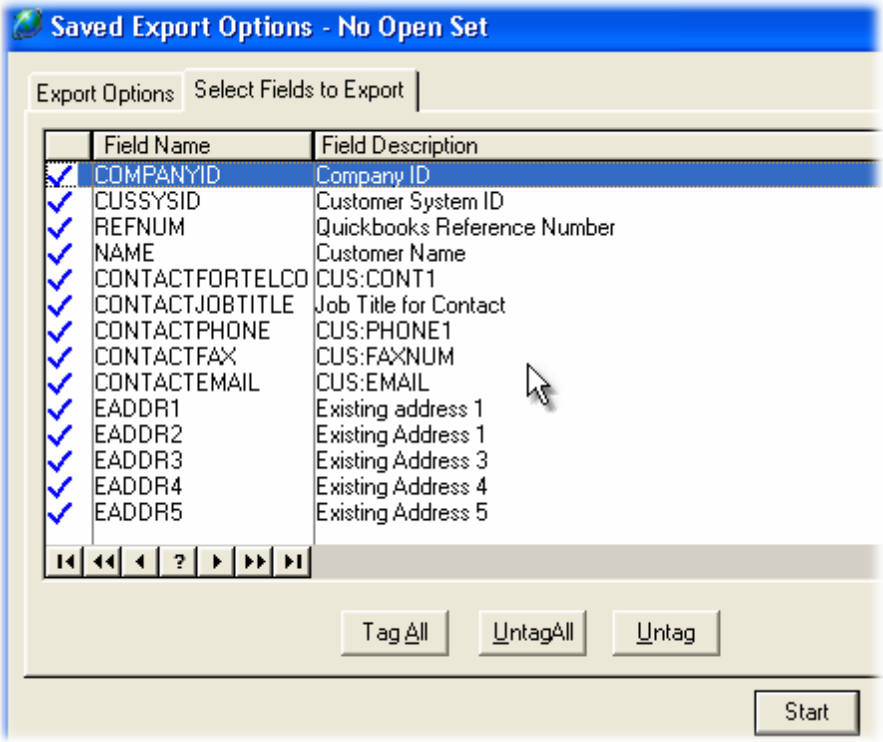
- Comma
- Tab

- Semicolon
- Pipe



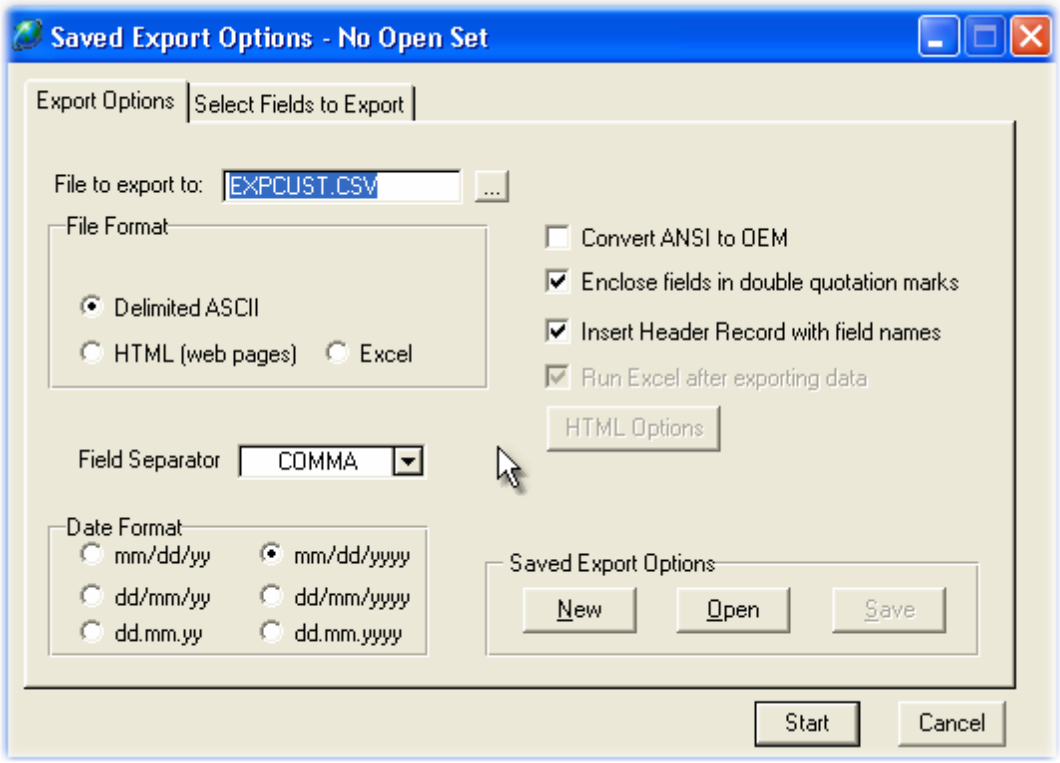

The HTML Option **radio button** will activate the **[HTML Options] button**.

The **Excel** radio button will activate the Header and **Run Excel option**s.

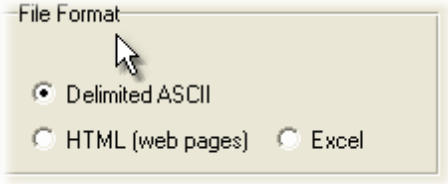



Fields may be **tagged** for export.

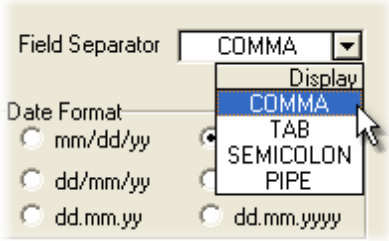
### 3.3.3 Exporting as Excel Files




**[File to export to] field**, use the **[...] elipse button** to navigate and select the file that you wish to import.



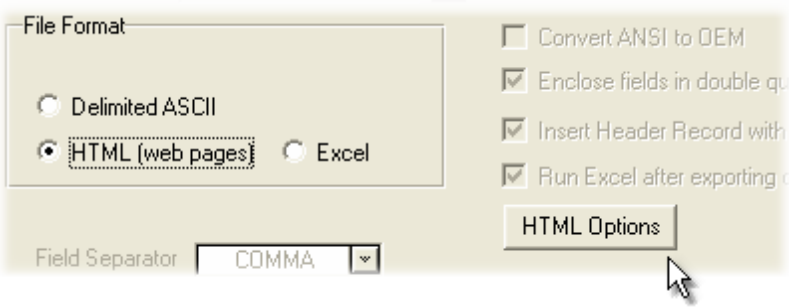

Select a **file format** that you wish to import.




Select from one of four **Field Separator**(s):

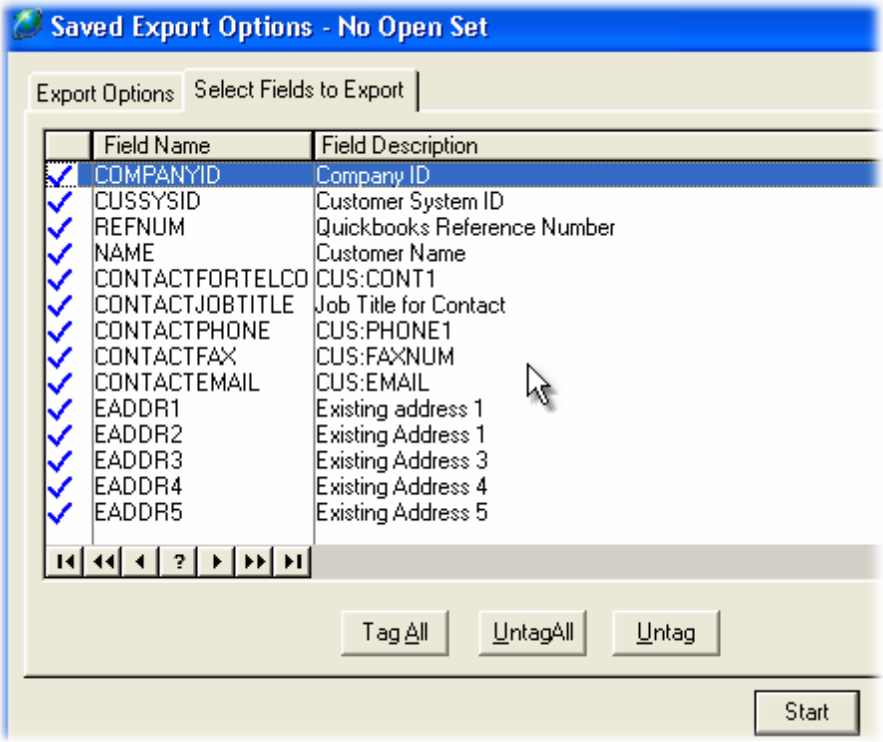
- Comma
- Tab

- Semicolon
- Pipe



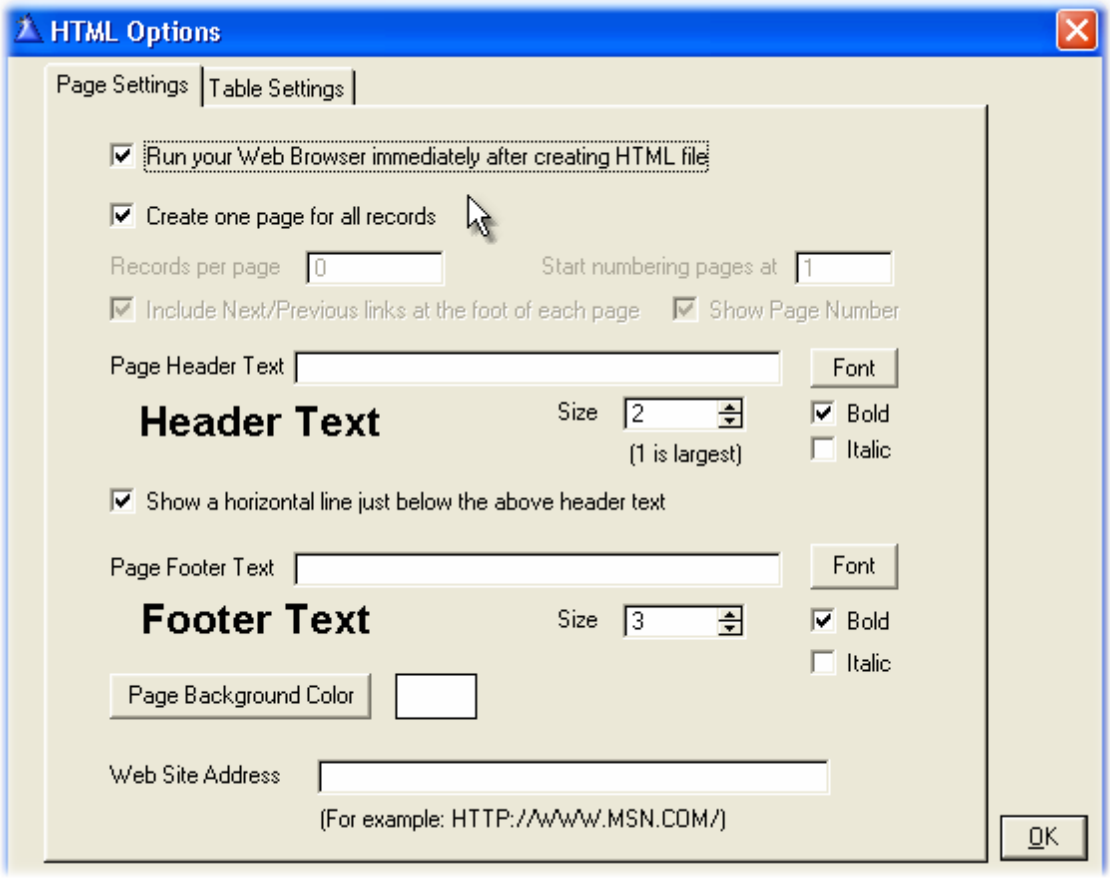

The HTML Option **radio button** will activate the **[HTML Options] button**.

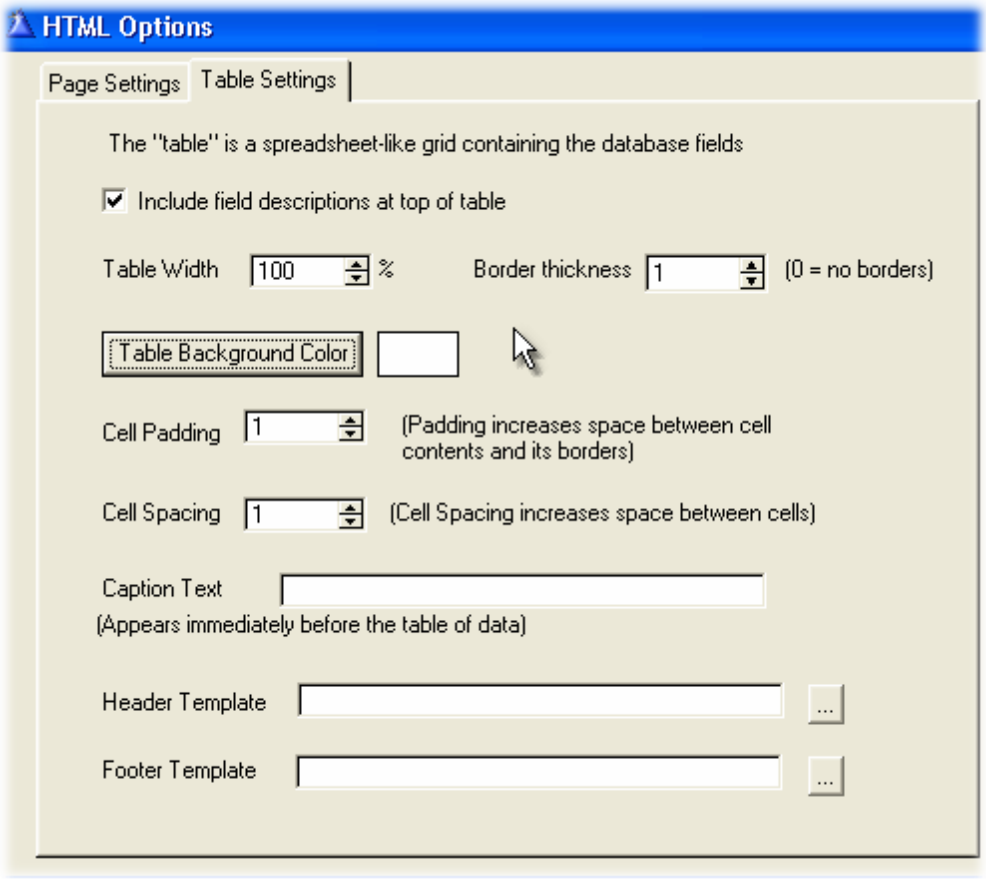
The **Excel** radio button will activate the Header and **Run Excel option**s.




Fields may be **tagged** for export.

### 3.3.4 HTML Options






## 3.4 Filtering an Export

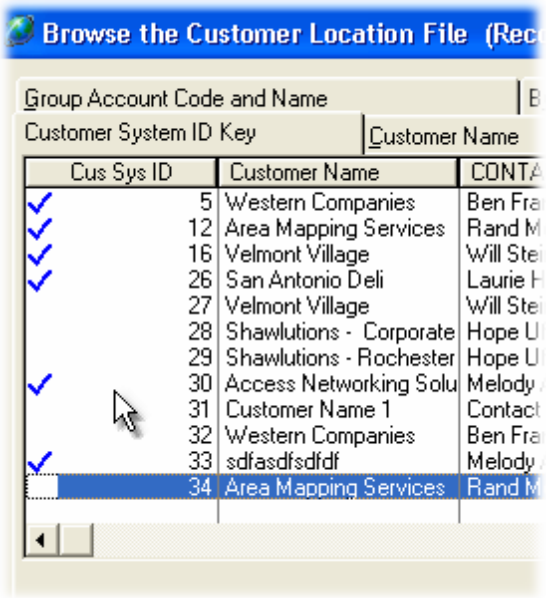

The **Export Process** can be **filtered** or export data can be **selected by tagging** the records that you wish to export.

A Browse Table is available for the following TelcoMgr files:

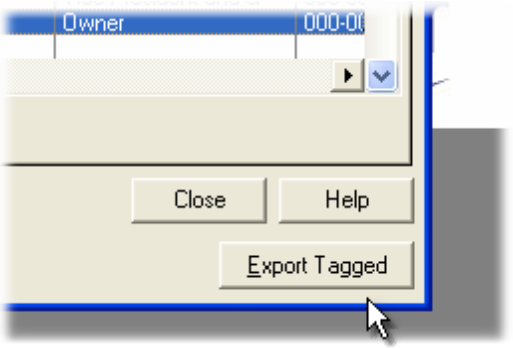
- Circuits File
- CPE Components
- Customer/Locations File
- Directory

- Expense Log
- History

### 3.4.1 Simple Tagging




Open of any of the **Browse Tables** and click on the appropriate line of the first column of each record.



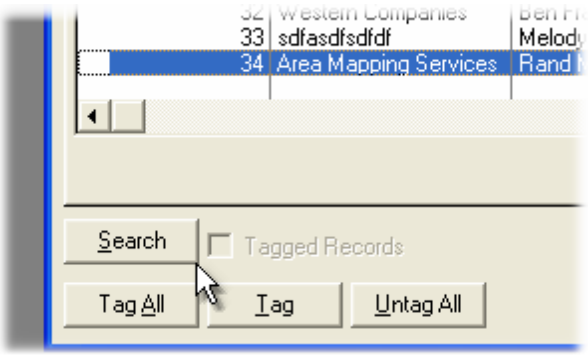

Select the **[Export Tagged] button**.

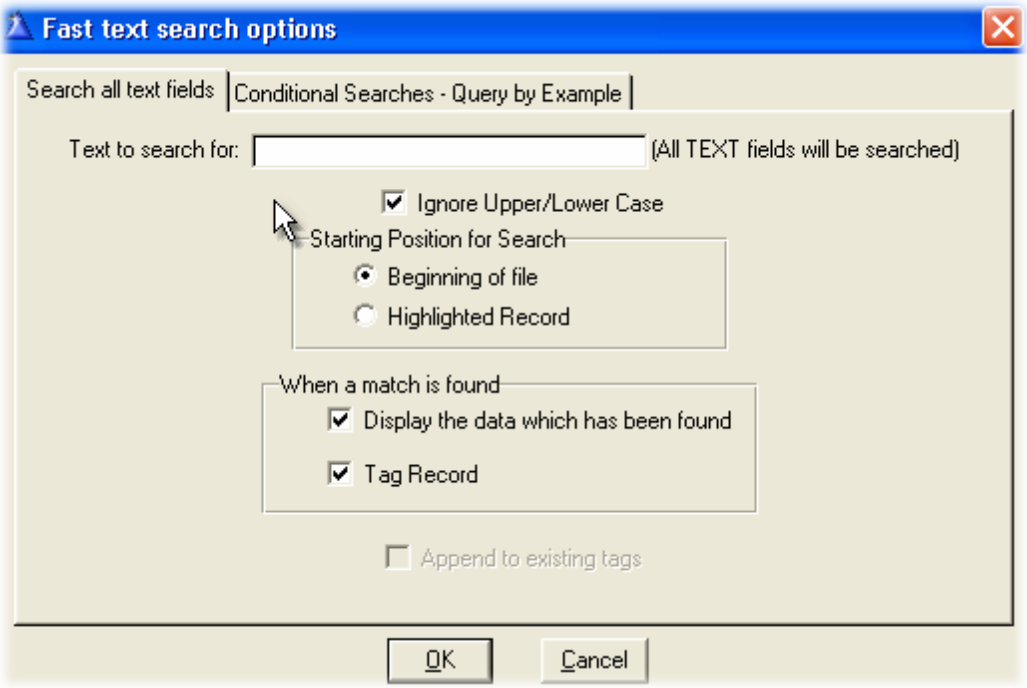
**Related Topics:**

Select Export Records by Query

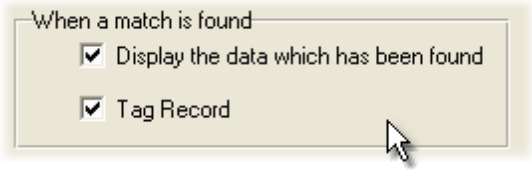
### 3.4.2 Global Search and Tag

Select the **[Search] button** to **query** the files for the appropriate qualifier that you wish to create for your export.

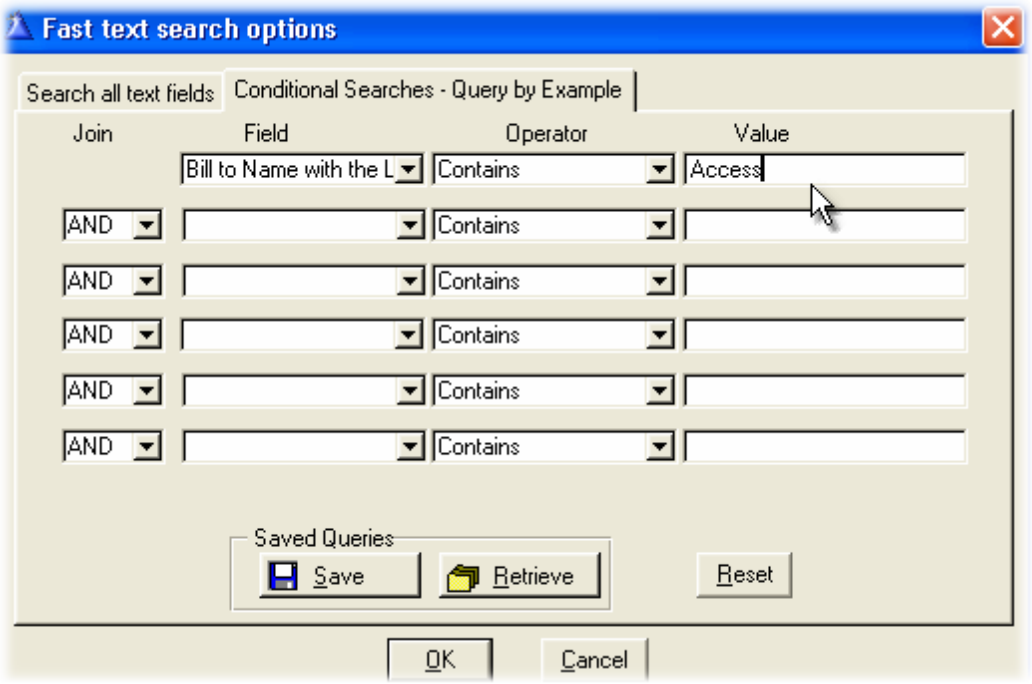



The **[Text to search for] field** is used to qualify for search; all records that equal the qualifier will be tagged when a match found.




Alternately, you can use a **Conditional Search** or **Query by Example**:




**Related Topic:**

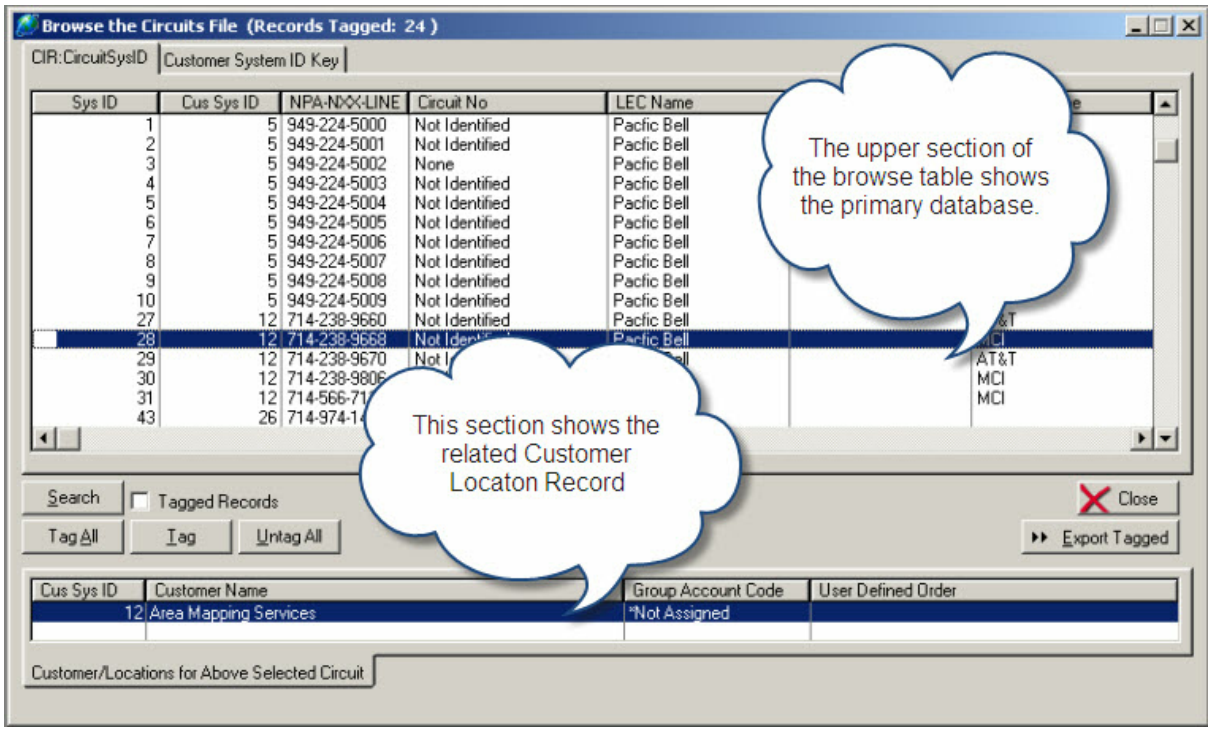
Select Export Records by Query

# 4 TelcoMgr Selected Export

Under the **Select Export Records** from the main menu you will find the following options:

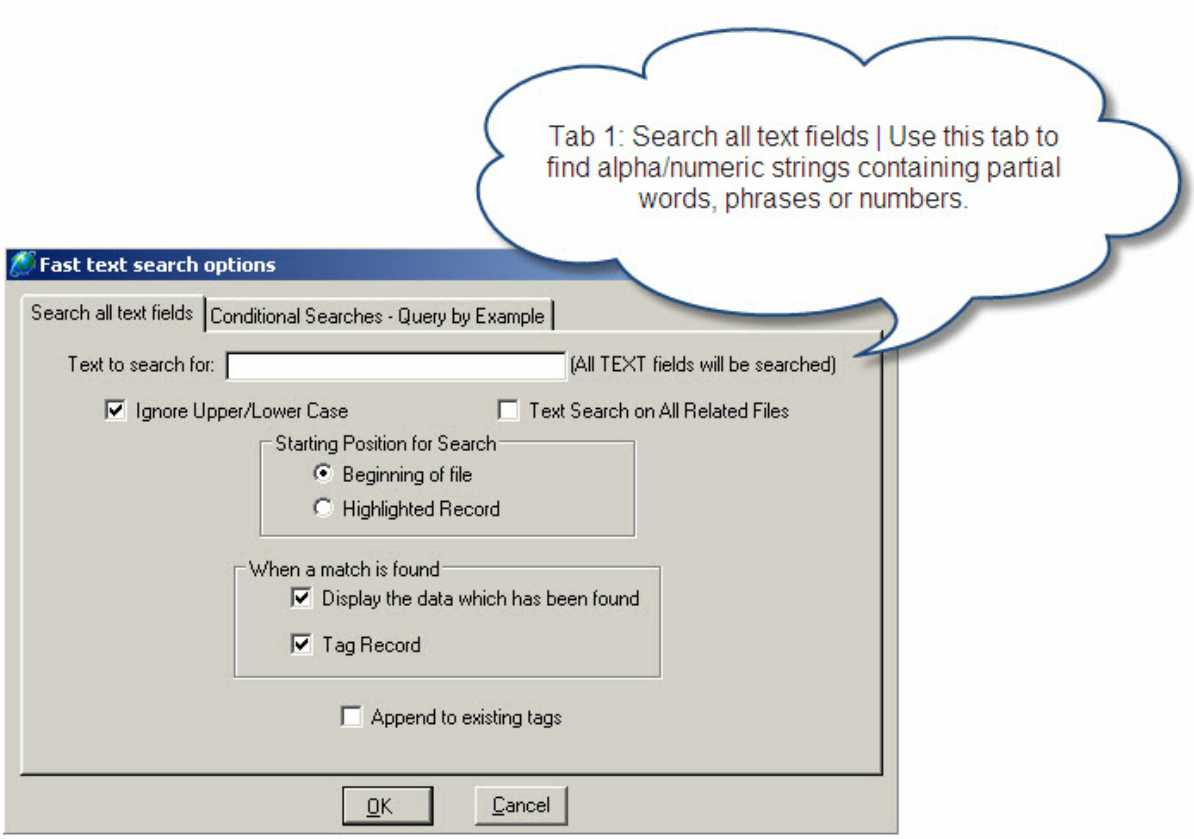

**User Tip:** Export a TelcoMgr Location, Circuits, CPE, History and/or Directory file as a **Comma Separated Value** (**CSV**) or (**Excel**)file to create a file template for importing data.

The above menu options provide a browse table query input window for locating and tagging records from the TelcoMgr databases.



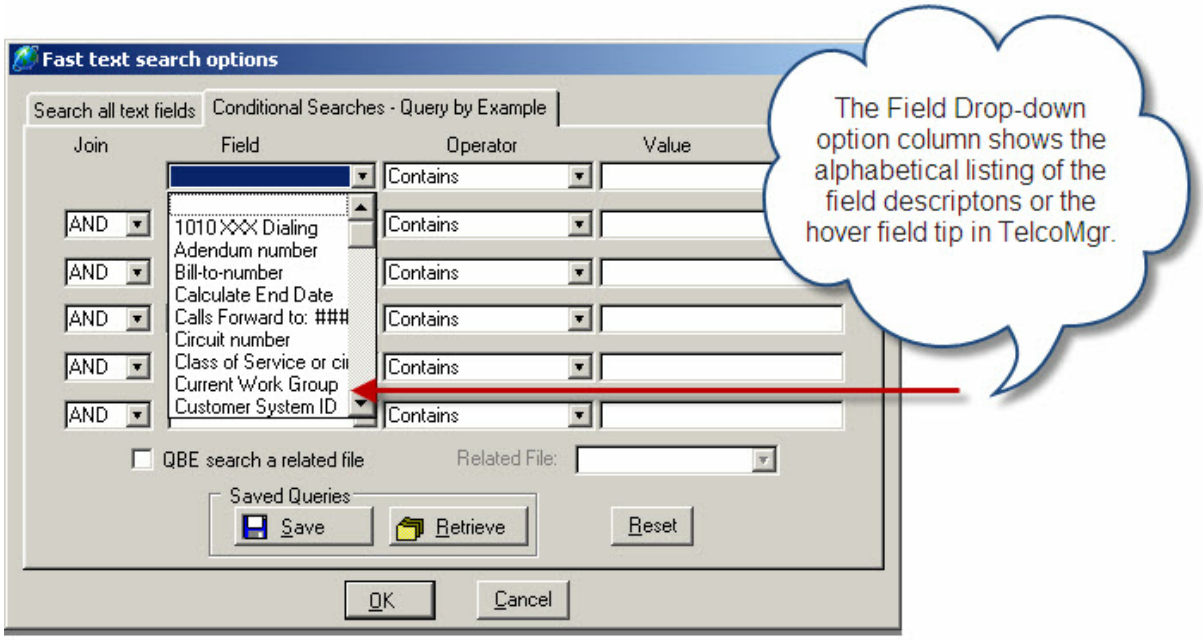

Click on the [**SEARCH**] button to start a query or to filter out the appropriate data for export.

TAB 1: **Search all text fields**



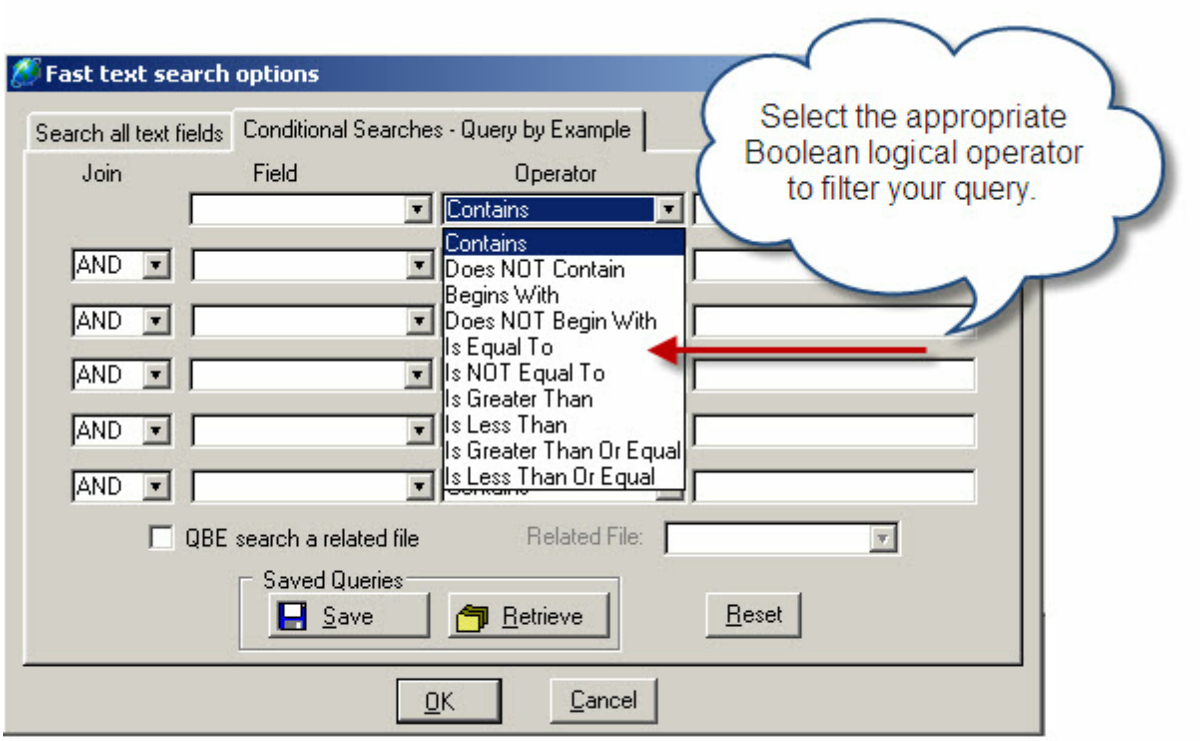

Use the [Text to search for:] field for general searches. Enter information that will be search across all text fields.
Click on the [X] Ignore Upper/Lower Case and [X] Text Search on All Related Files to broaden or narrow your search.
[X] Append to existing tags allows you to "add to" previously tagged records.

**TAB 2: Conditional Searches - Query by Example**

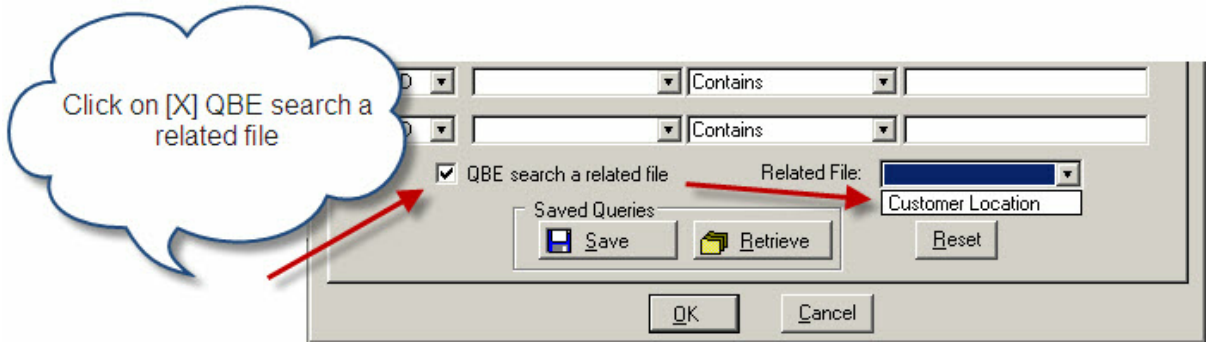



**Tab 2** provides search options that may be combined with the **Tab 1** '**string search options'** or without an entry in the string search field. Also, you may combine the search on six fields simultaneously.

**USER TIP:** Start your query with the least amount of data to query and then modify your query if necessary after you have evaluated your results.



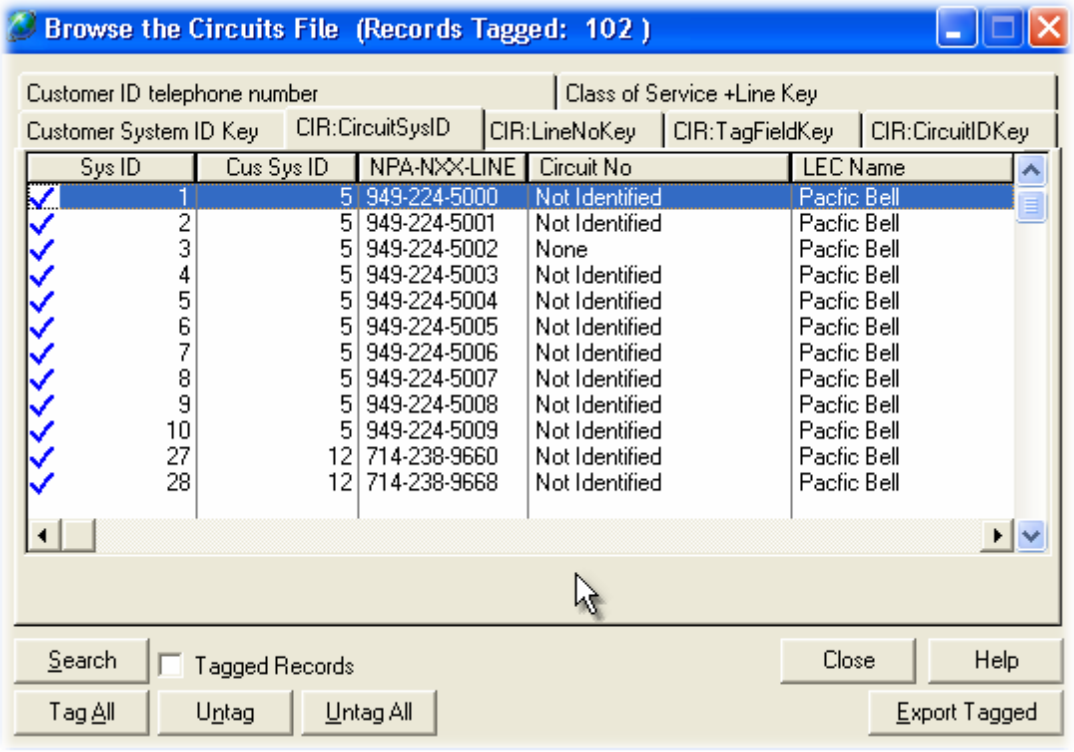

A **Boolean Logically evaluation** may be executed on each field that you select.




Optionally, you may search or execute a query on the **Related Files**.

**USER TIP:** Experiment with your query and evaluate the results by using the Tagged Browse table. Also, run several practice Exports before settling on your final results.

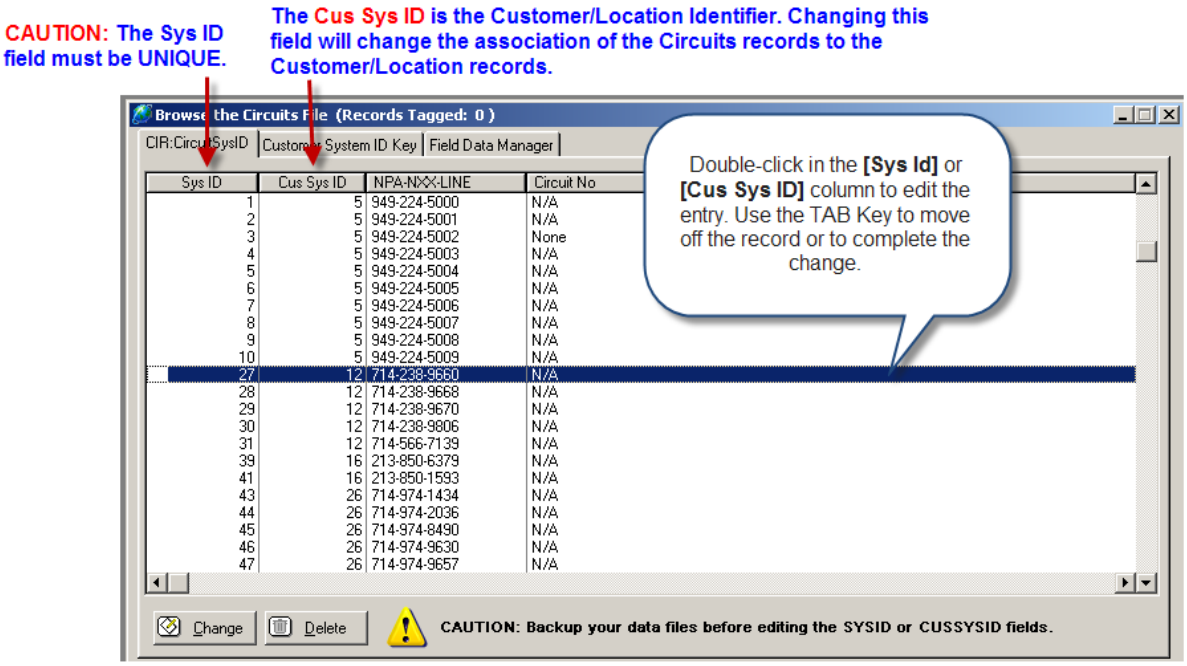
## 4.1 Browse Circuits



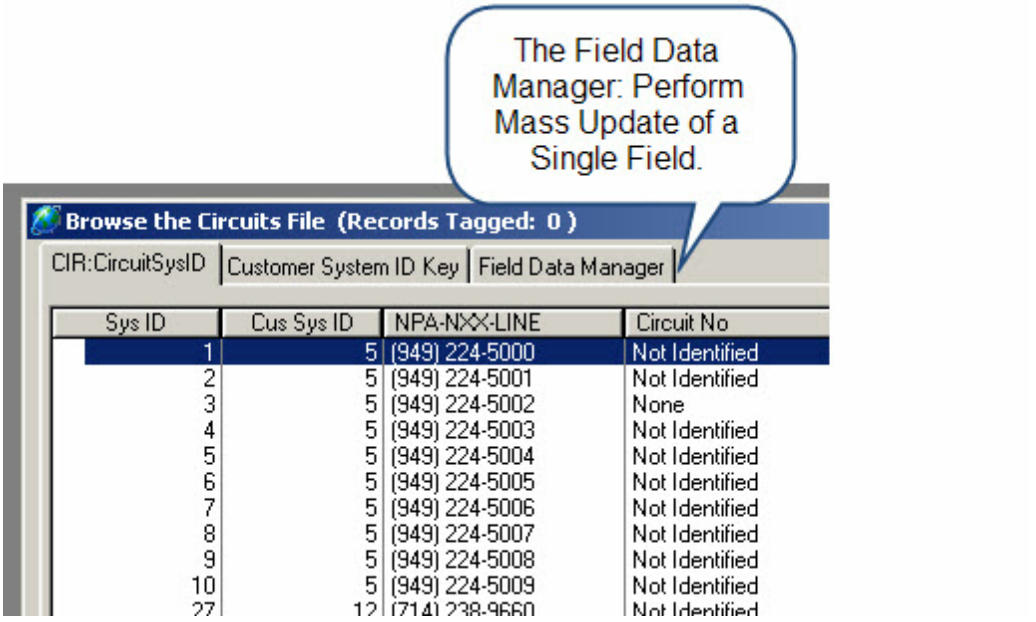

Use the various browse tables to evaluate your query results.

**Editing the Customer/Location Identifier [CusSysId] for the Circuits file.**

The **[CusSysId]** for the Circuit record must match a Customer/Location SysID to appear inside the Circuits Tab of the Customer record. If you enter a number in the **[CusSysId]** field that does not appear in the Customer/Location file as a SysId then the Circuit records will become orphaned or unassociated with a Customer/Location.




Highlight the record to be edited and click on the [**Change**] button. Use the **TAB Key** to move to the [**Cus Sys ID**] field.




**CAUTION:** The Field Data Manager is an advanced option.


## 4.2 Field Data Manager

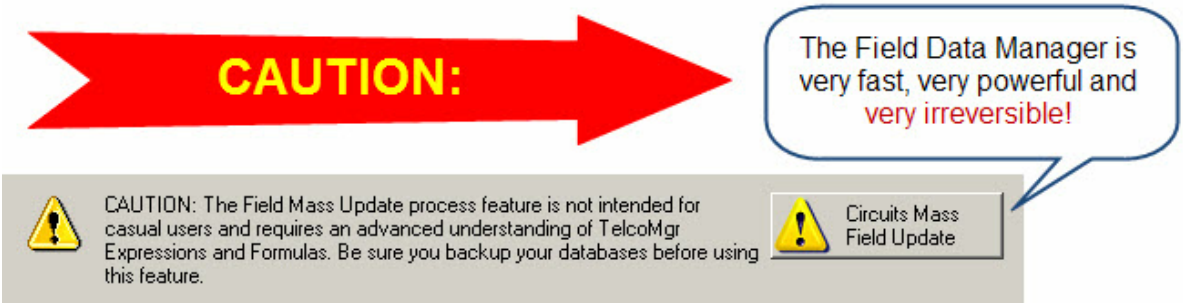
A **Field Data Manager** for the **Customer/Location** and **Circuits** file is available from the Browse Customer Table and Browse Circuits Table.


How many times have you been asked to make a mass update like change the format of the NPA-NXX-Line field for the Circuits table? Our new **Field Data Manager** provides end-users the possibility to alter the data as much as you wish. Thousands of phone numbers can be changed from the standard TelcoMgr format of [###-###-####] to [ +1 (###) ###-#### ] within a couple of minutes. Also, by using Boolean expressions, data changes may be applied to records that belong to a specific customer group or your User-Defined field. You can even store mass update routines for future use.

Use caution when using the **Field Data Manager**

because some of the changes that you apply may not be reversible. Therefore, backup your TelcoMgr folders before attempting to use this feature. Also, a strong understanding of how to use **TelcoMgr Expressions** and **Formulas** is essential to use this feature.

"The Field Data Manager is very fast, very powerful and very irreversible!"




### 4.2.1 Field Data Manager Expressions

**Expressions** are evaluated in the standard algebraic order of operations. The precedence of operations is controlled by **operator** type and placement of parentheses. Each operation produces an (internal) intermediate value used in subsequent operations. Parentheses may be used to group operations within expressions. Expressions are evaluated beginning with the inner-most set of parentheses and working through to the outer-most set.
Precedence levels for expression evaluation, from highest to lowest, and left-to-right within each level, are:

| | | |
|---|---|---|
| Level 1 | ( ) | Parenthetical Grouping |
| Level 2 | - | Unary Minus (Negative sign) |
| Level 3 | procedure call | Gets the RETURN value |
| Level 4 | ^ | Exponentiation |
| Level 5 | * / % | Multiplication, Division, Modulus Division |
| Level 6 | + - | Addition, Subtraction |
| Level 7 | & | Concatenation |
| Level 8 | = <> | Logical Comparisons |
| Level 9 | NOT, AND, OR/XOR | Boolean expressions |

Level 10 Single quotes (apostrophe) 'myexpression' Used for absolute data pictures

Expressions may produce numeric values, string values, or logical values (true/false evaluation). An expression may contain no operators at all; it may be a single variable, constant value, or procedure call which returns a value.

#### Sting Constants

A string constant is a set of characters enclosed in single quotes (apostrophes). The maximum length of a string constant is 255 characters. Characters that cannot be entered from the keyboard may be inserted into a string constant by enclosing their ASCII character codes in angle brackets (<> ). ASCII character codes may be

represented in decimal, hexadecimal, binary, or octal numeric constant format. In a string constant, a left angle bracket ( < ) initiates a scan for a right angle bracket. Therefore, to include a left angle bracket in a string constant requires two left angle brackets in succession. To include an apostrophe as part of the value inside a string constant requires two apostrophes in succession. Two apostrophes ( '' ), with no characters (or just spaces) between them, represents a null, or blank, string. Consecutive occurrences of the same character within a string constant may be represented by repeat count notation. The number of times the character is to be repeated is placed within curly braces ( { } ) immediately following the character to repeat. To include a left curly brace ({) as part of the value inside a string constant requires two left curly braces ({{) in succession.

The **ampersand** (&) is always valid in a string constant. However, depending on the assignment's destination, it may be interpreted as an underscore for a hot letter (for example, a **PROMPT** control's display text). In this case, you double it up (&&) to end up with a single ampersand in the screen display.

Example:

```
'string constant'   !A string constant

'It''s a girl!'     !With embedded apostrophe
'<27,15>'           !Using decimal ASCII codes
'A << B'            !With embedded left angle, A < B
'*{20}'             !Twenty asterisks, repeat-count notation
''                  !A null (blank) string
```

**Data Formatting** or **Pictures**

@P All pattern pictures begin with the @P delimiter and end with the P delimiter. The case of the delimiters must be the same.

< Specifies an integer position that is blank for leading zeroes.
# Specifies an integer position.
x Represents optional display characters. These characters appear in the final result string.
P All pattern pictures must end with P. If a lower case @p delimiter is used, the ending P delimiter must also be lower case.
B Specifies that the format displays as blank when the value is zero.

Pattern pictures contain optional integer positions and optional edit characters. Any character other than < or # is considered an edit character which will appear in the formatted picture string. The @P and P delimiters are case sensitive. Therefore, an upper case "P" can be included as an edit character if the delimiters are both lower case "p" and vice versa.
Pattern pictures do not recognize decimal points, in order to permit the period to be used as an edit character. Therefore, the value formatted by a pattern picture should be an integer. If a floating point value is formatted by a pattern picture, only the integer

portion of the number will appear in the result.

Example:

| Picture | Value | Result |
|---|---|---|
| @P###-##-####P | 215846377 | 215-84-6377 |
| @P<#/##/##P | 103159 | 10/31/59 |
| @'P(###)###-####P' | 3057854555 | (305)785-4555 |
| @P###/###-####P | 7854555 | 000/785-4555 |
| @p<#:##PMp | 530 | 5:30PM |
| @P<#' <#"P | 506 | 5' 6" |
| @P<#lb. <#oz.P | 902 | 9lb. 2oz. |
| @P4##A-#P | 112 | 411A-2 |
| @PA##.C#P | 312.45 | A31.C2 |

### 4.2.2 Field Data Manager Operators

#### Logical Operators

A **logical operator** compares two operands or **expressions** and produces a true or false condition. There are two types of logical operators: conditional and Boolean. Conditional operators compare two values or expressions. Boolean operators connect string, numeric, or logical expressions together to determine true-false logic. Operators may be combined to create complex operators.

Conditional Operators

| | |
|---|---|
| = | Equal sign |
| < | Less than |
| > | Greater than |

Boolean Operators

| | |
|---|---|
| NOT | Boolean (logical) NOT |
| ~ | Tilde (logical NOT) |
| AND | Boolean AND |
| OR | Boolean OR |
| XOR | Boolean eXclusive OR |

Combined operators

| | |
|---|---|
| <> | Not equal |
| ~= | Not equal |
| NOT = | Not equal |
| <= | Less than or equal to |
| =< | Less than or equal to |
| ~> | Not greater than |
| NOT > | Not greater than |
| >= | Greater than or equal to |
| => | Greater than or equal to |
| ~< | Not less than |
| NOT < | Not less than |

During logical evaluation, any non-zero numeric value or non-blank string value indicates a true condition, and a null (blank) string or zero numeric value indicates a false condition.

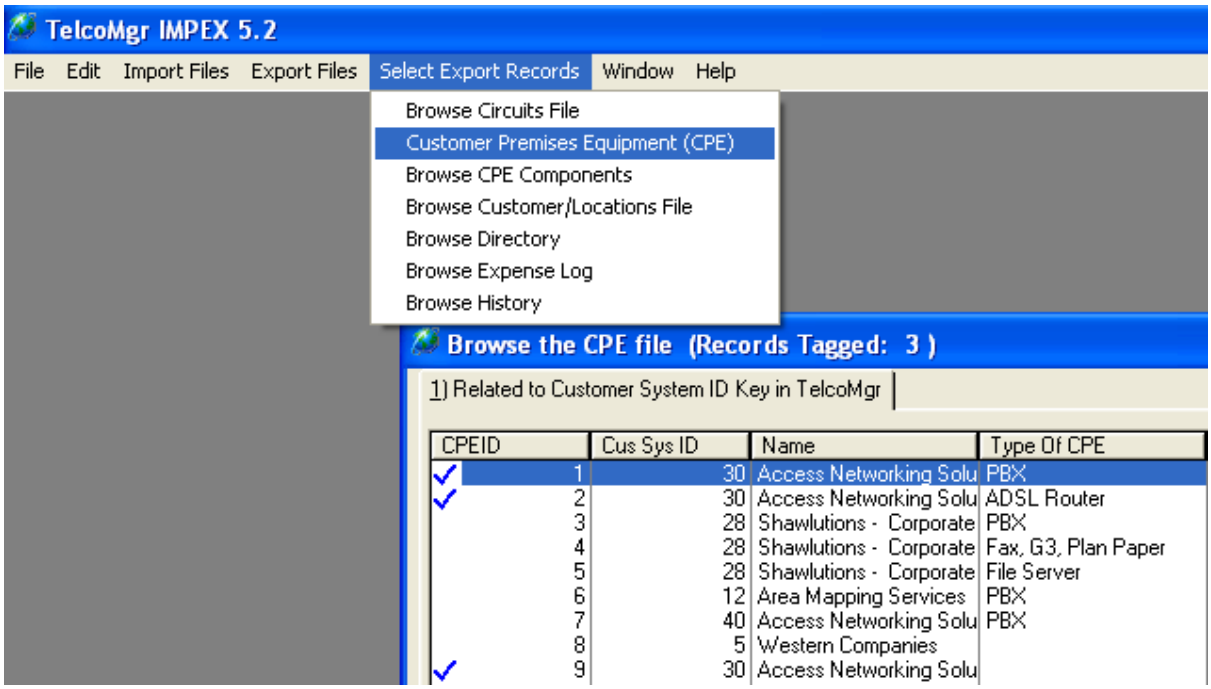
Example:

| Logical Expression | Result |
|---|---|
| A = B | True when A is equal to B |
| A < B | True when A is less than B |
| A > B | True when A is greater than B |
| A <> B, A ~= B, A NOT = B | True when A is not equal to B |
| A ~< B, A >= B, A NOT < B | True when A is not less than B |
| A ~> B, A <= B, A NOT > B | True when A is not greater than B |
| ~ A, NOT A | True when A is null or zero |
| A AND B | True when A is true and B is true |
| A OR B | True when A is true, or B is true, or both are true |
| A XOR B | True when A is true or B is true, but not both. |

## 4.3 Browse CPE Table

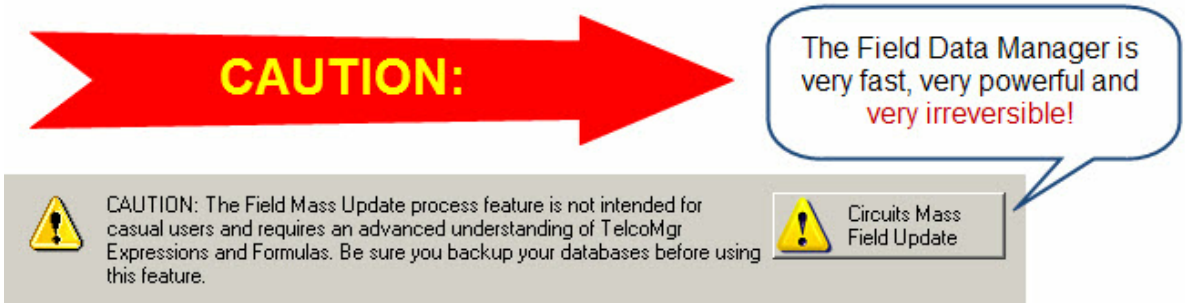
The Customer/Location record is the PARENT for the Customer Premises Equipment table, which has a CHILD table of more specific components and called the **CPE Components** table.

The image below shows the CPE table:

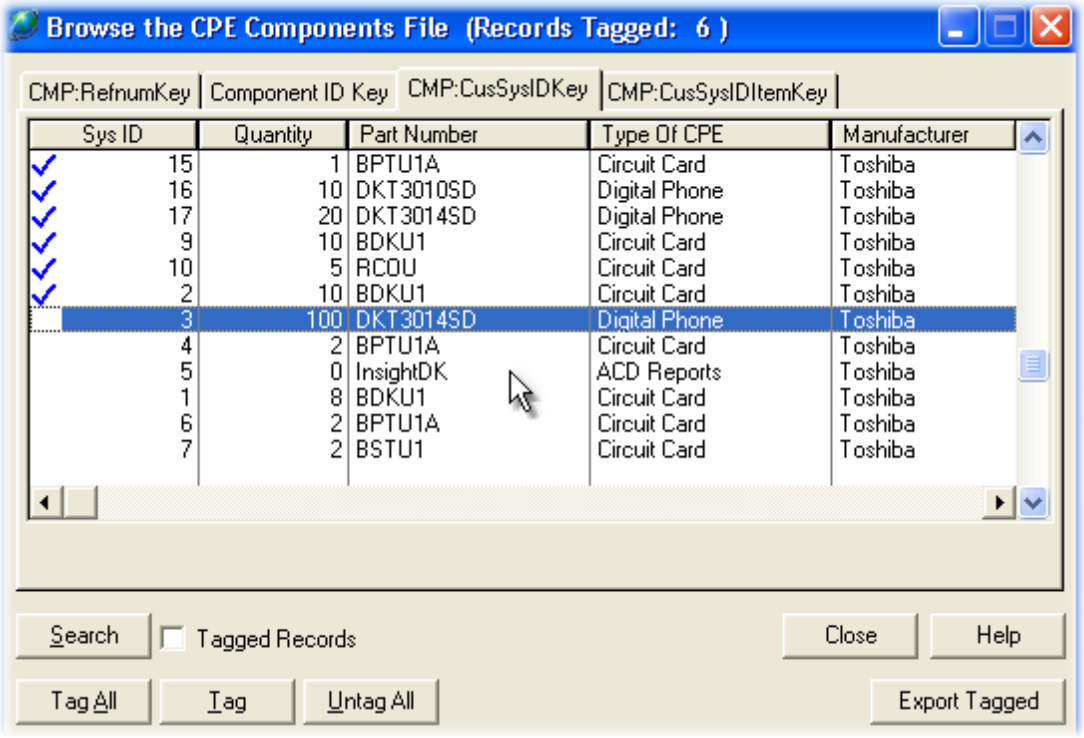



In the example above, PBX generically describes a collection of equipment, including telephones, servers and cabinets. Therefore, more detailed components may be stored in the Components table.

"The Field Data Manager is very fast, very powerful and very irreversible!"

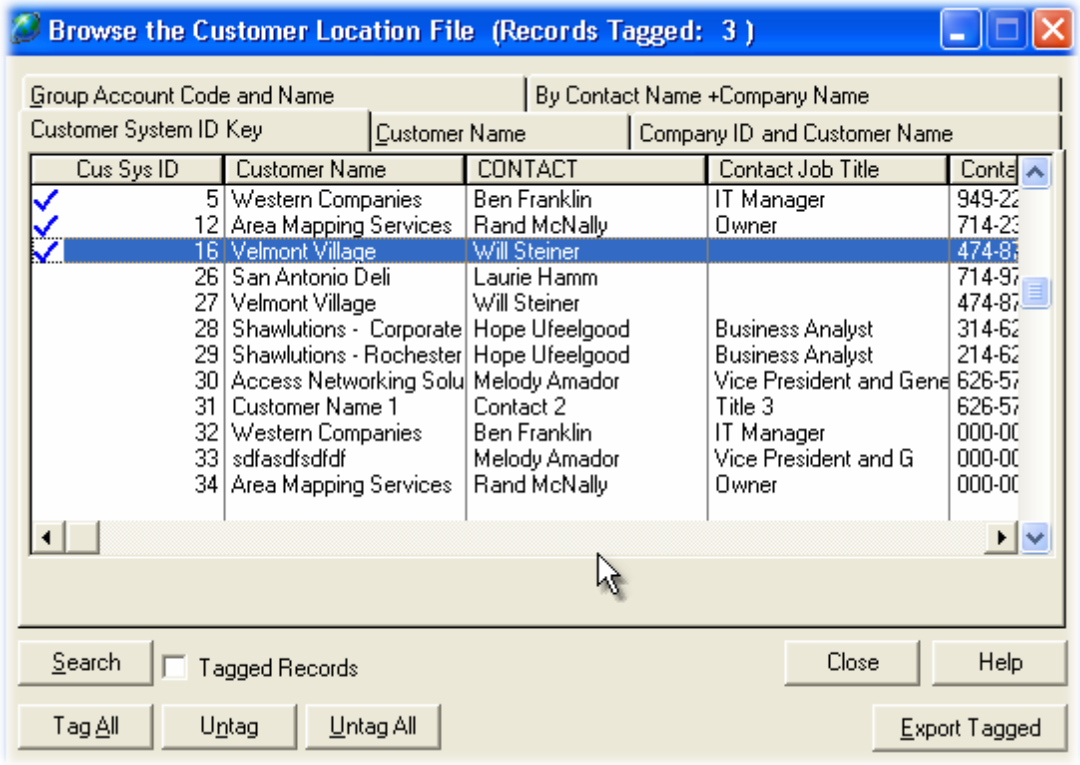


### 4.3.1 Browse CPE Components

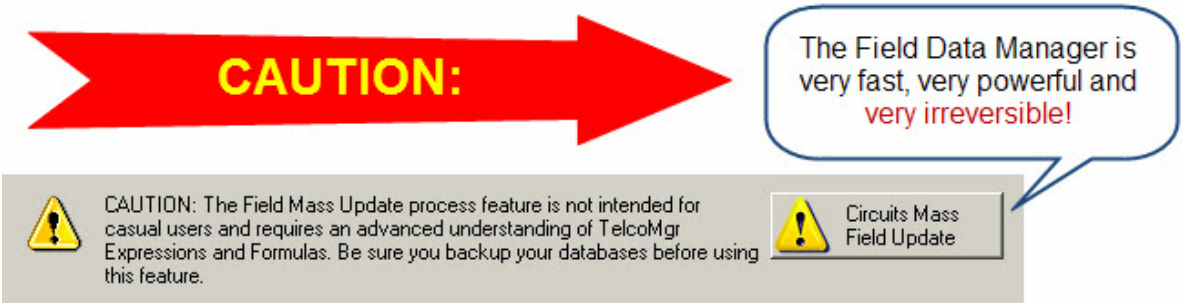



The CPE Components table is a sub-file or CHILD database of the CPE Table.

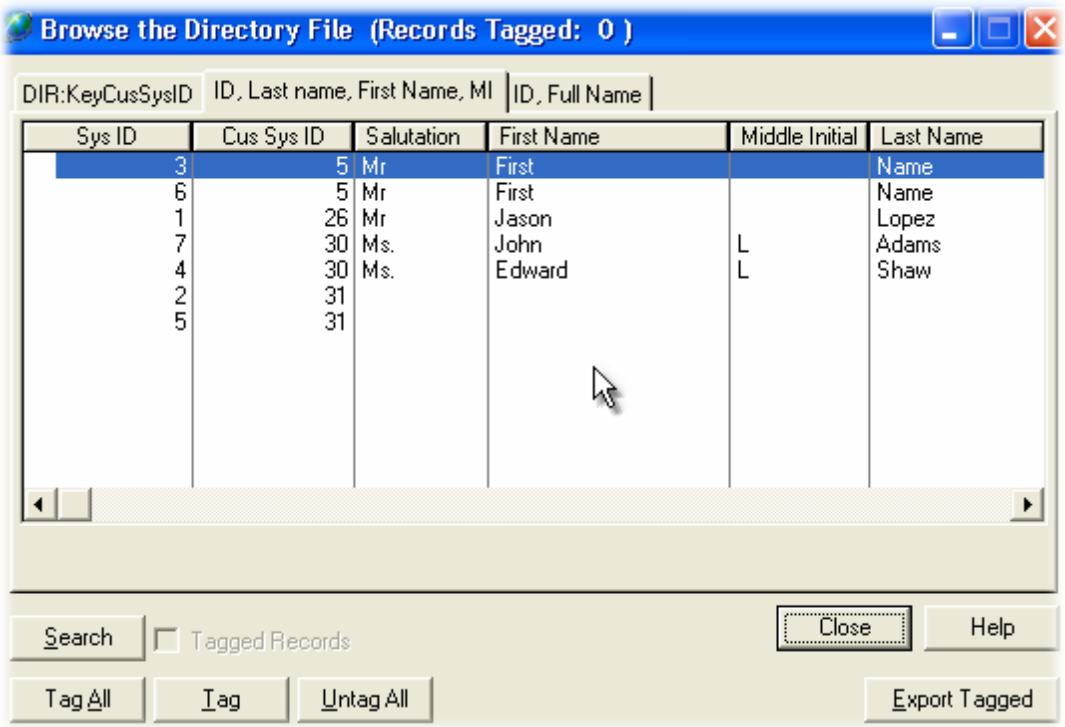
## 4.4 Browse Customers



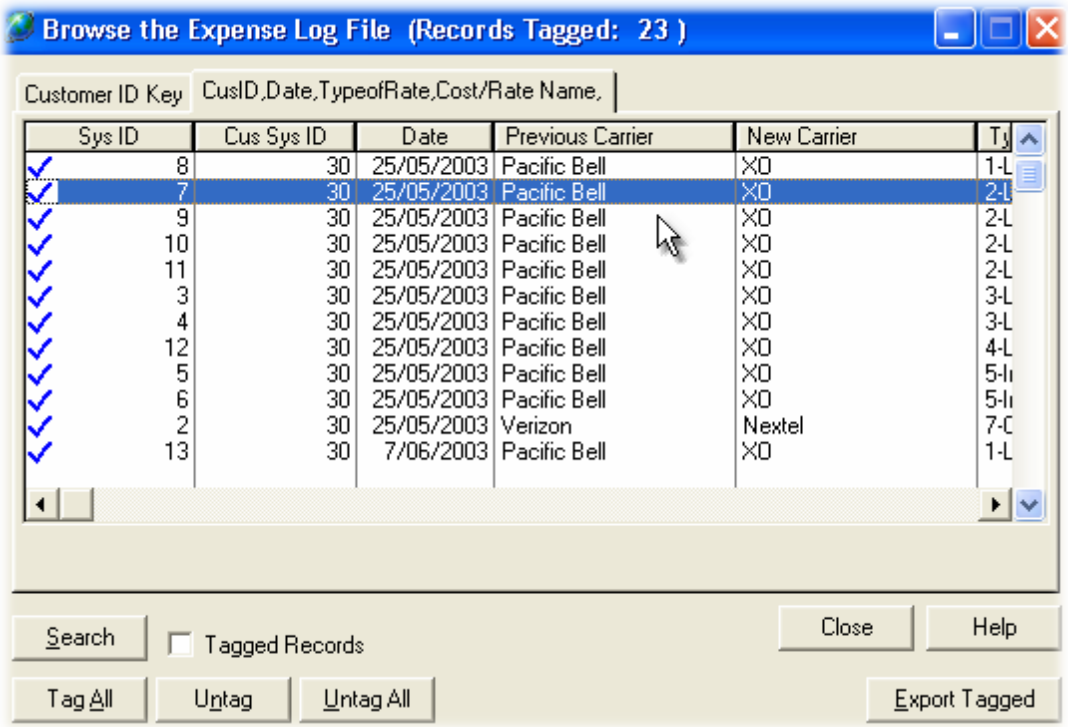

"The **Field Data Manager** is very fast, very powerful and very irreversible!"



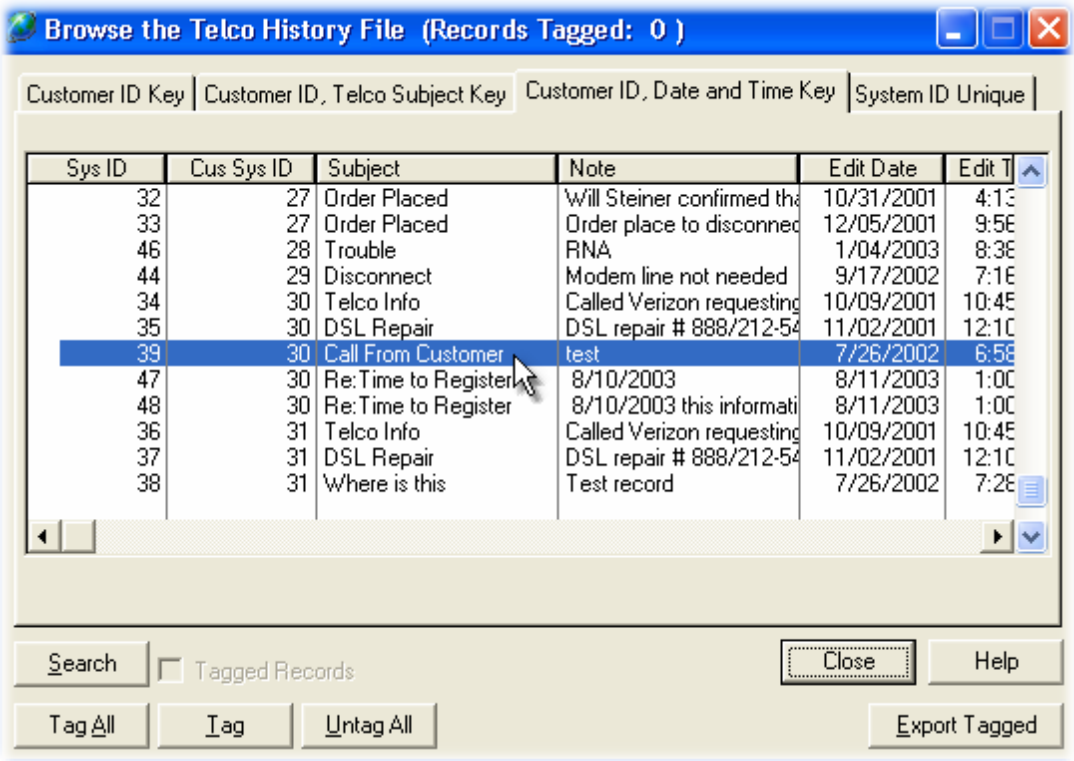
## 4.5 Browse Directory



## 4.6 Browse Expense Log



## 4.7 Browse History



# Index

## - [ -



## - A -



## - B -



## - C -



## - D -



## - E -



## - F -



## - G -

# - H -



# - I -



# - L -



# - M -



# - N -



# - O -



# - P -



# - Q -



# - R -



# - S -



# - T -



# - U -



# - V -

# - W -