: ELECTRONIC EXCHANGE

Introduction

What is the Electronic Exchange? It is a mechanism for Data Exchange between INFOLAB and other systems, which may also be INFOLAB, or other.

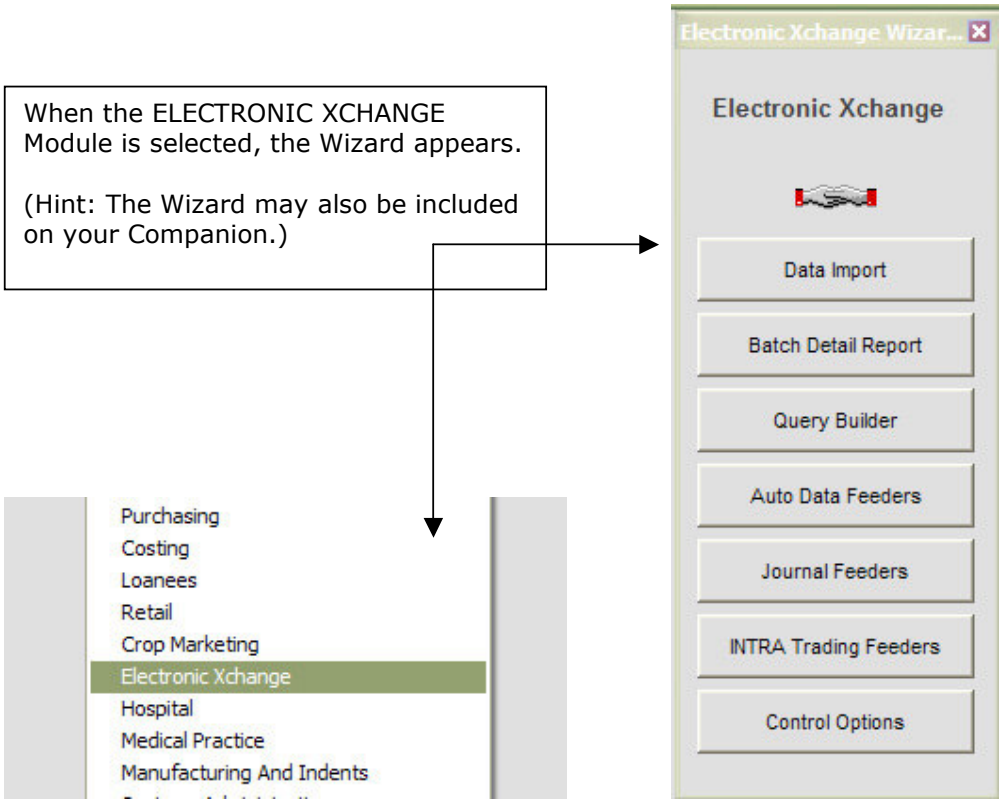
There are 3 primary types of Data Exchange: -

1) Automatic Data Feeders
2) Journal Feeders
3) INTRA Trading Exchange

The purpose of this User Manual is to explain what these 3 methods are, and how they can be used in INFOLAB.

The Electronic Exchange Wizard

The Electronic Exchange Wizard appears when you select this Module from the Menu system.



The options 'Data Import', 'Batch Detail Report' and 'Query Builder' are simply useful links to standard INFOLAB options that are also used elsewhere. They are

included here for convenience, since they are Functions that you will use with some of the Feeder sequences.
Although these 3 options are explained in detail in their 'home' User Manuals, we will just indicate here briefly what they are for.

DATA IMPORT is used to import Data from a File outside INFOLAB, into an INFOLAB accessible format.

BATCH DETAIL REPORT produces a Report with detail of all Transactions included in that Batch, and is used here for checking results of imported and posted Journals.

QUERY BUILDER is an INFOLAB function that exports Data from INFOLAB to a File of your choice, and which may then be used by another system.

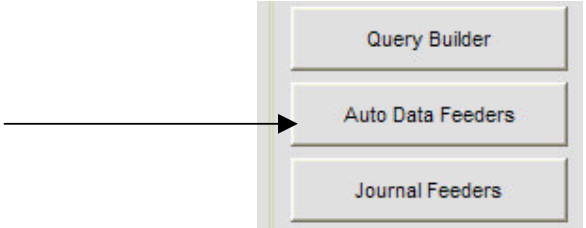
After these 3 choices on the Wizard shown above, we have the FEEDER choices, i.e. AUTO DATA FEEDERS, JOURNAL FEEDERS and INTRA TRADING FEEDERS. Once a User has access to any of these Feeders, the same User also has access to all the options within that Feeder. However, in the case of Journal Feeders, the Journal Feeder Definition Record indicates the list of Users who may execute that specific Feeder, and therefore the Definition option cannot be included among the JOURNAL FEEDER options. It has therefore been provided as an option with CONTROL OPTIONS (the last option on the Wizard), in order to control access to the option.

And now we may have a look at each feeder type, one after the other.

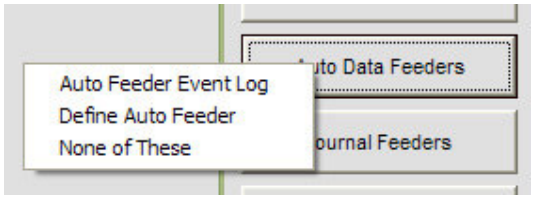
AUTO DATA FEEDERS

Auto Data Feeders are Feeders that are specially set up to synchronize data between INFOLAB and another system, and they may have the capability also to generate Journals in INFOLAB. An example of this is a system that performs a specialized function, e.g. Loan Originations, with INFOLAB performing the backend. In such an example, there are of necessity a number of data tables that are 'shared', e.g. Client Master, Loan Master, etc. When the one system creates a new Record on such a table, or amends a Record, it needs to advise the other system. In addition, the other system may even perform some financial Transactions, e.g. a Cheque or Receipt, and needs to advise INFOLAB of each such event. An Auto Data Feeder then, is a program that will automatically process inputs from another system (usually without any User intervention), and that may also generate outputs to be processed by the other system.

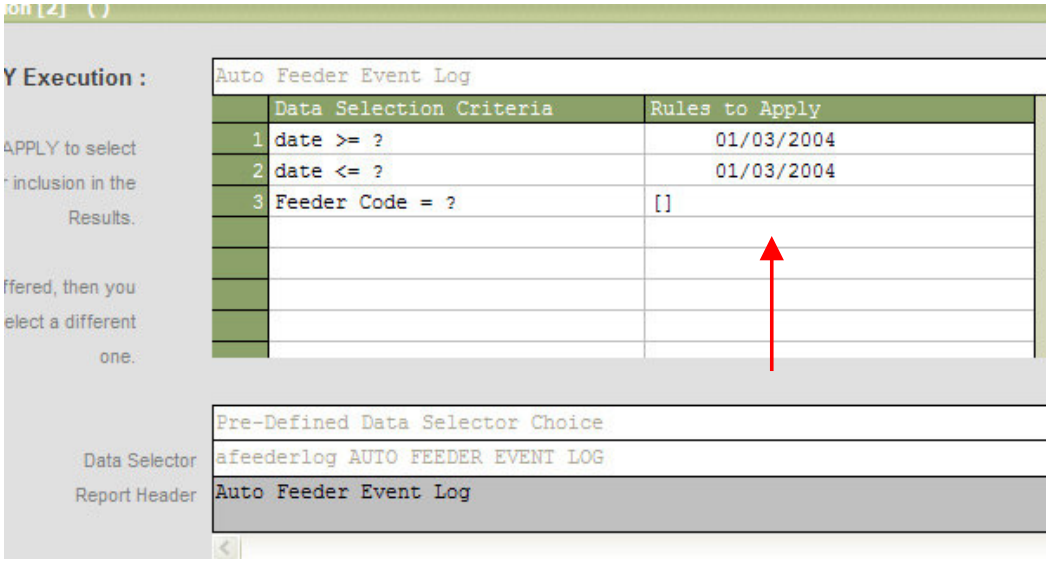
Auto Data Feeders can only be designed and implemented by experienced INFOLAB Developers. Once implemented, they are easy to use. We will now explain how to use the inherent ILS (INFOLAB LOAN SERVICES) Auto Data Feeder, and the same principles should apply to any other Auto Data Feeders that you may use in INFOLAB.



Once we choose AUTO DATA FEEDERS, we are offered the following options –



'Auto Feeder Event Log' is the most often used option, i.e. to verify the Log in order to check for any abnormalities.

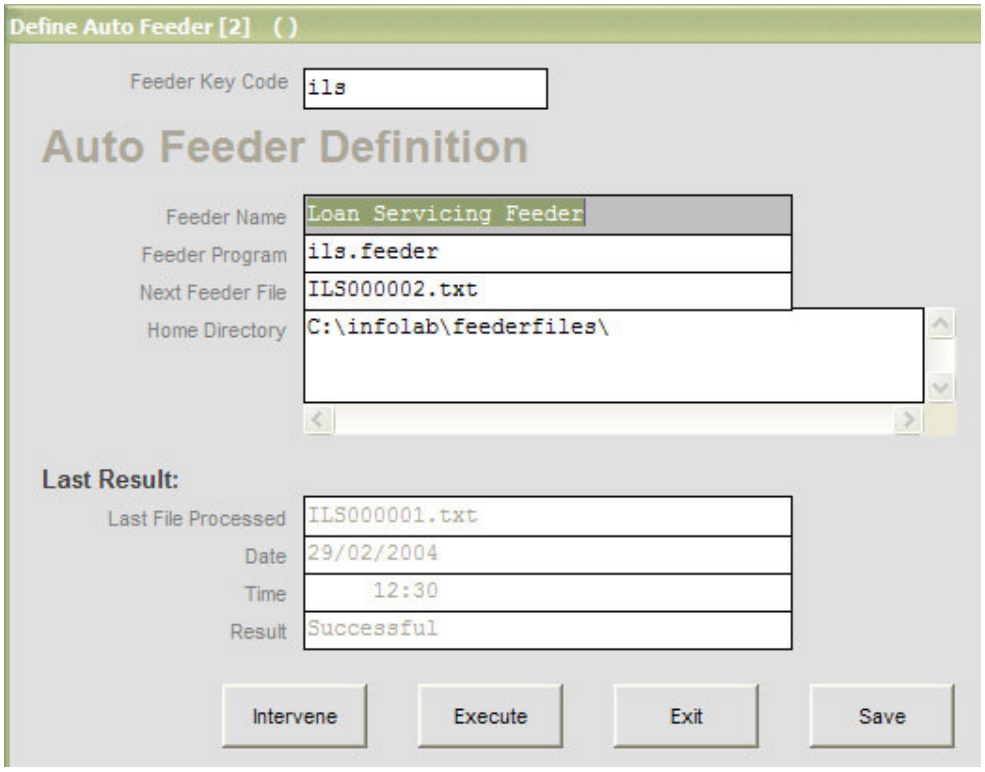


The clip shown above is from the standard Report Template Screen when executing a Report, in this case the 'Auto Feeder Event Log'. Note that the Report may be produced for any or a specific Feeder Code, and that you can specify a Date Range as to which Events will be included in the Report.

```
08:38:26 01 Mar 2004 page    1

 Auto Feeder Event Log


Feeder. Date..... Time..... User.............. Input File..... Description.............

ILS     29/02/2004      12:18 System Manager    ILS000001.txt   Start New File Import
ILS     29/02/2004      12:26 System Manager    ILS000001.txt   In Progress
ILS     29/02/2004      12:30 System Manager    ILS000001.txt   Successful Update

[405] 3 items listed out of 3 items.
```

In the example shown above we can see 'normal' Feeder Events. We can see that a Run was started, was 'in progress', and concluded successfully. If a Feeder reports problems and cannot continue (very unlikely), then we may choose the INTERVENE option shown in the next example.

'Define Auto Feeder' is the other option offered with Auto Data Feeders, and it looks like this.

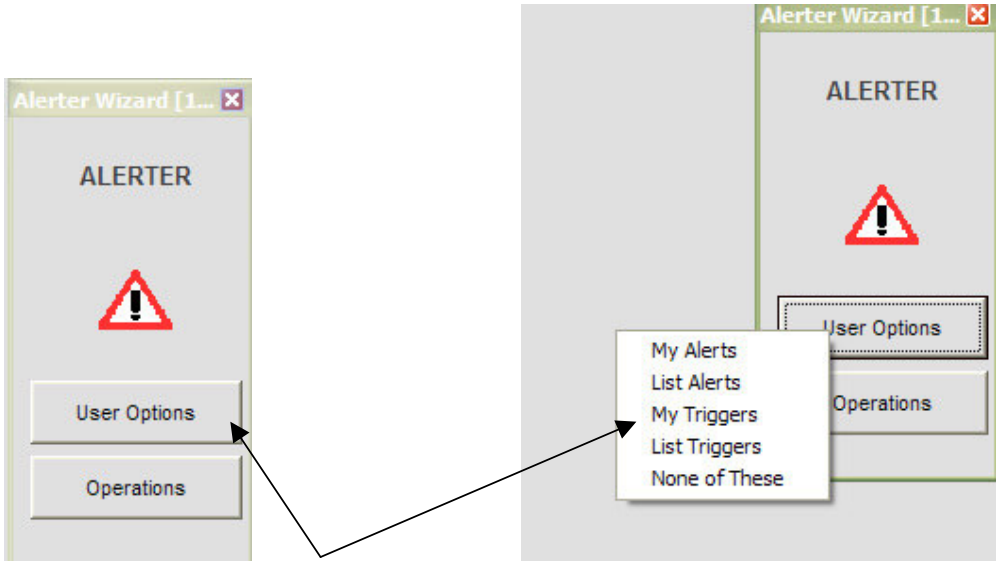


An Auto Data Feeder has a Code, a Name, and a very specific Program that operates it. Other than that, there is always a 'next' Feeder File that is expected, and a Home Folder where INFOLAB should look for Data to Import. When a Feeder is 1st implemented, the 'next' File is specified, but after that, the Feeder will update this Field by itself. Only in rare circumstances (a hiccup on the originating side?) it may be necessary to skip a File in the range, in which case it will be necessary to 'feed' the next File to expect here.

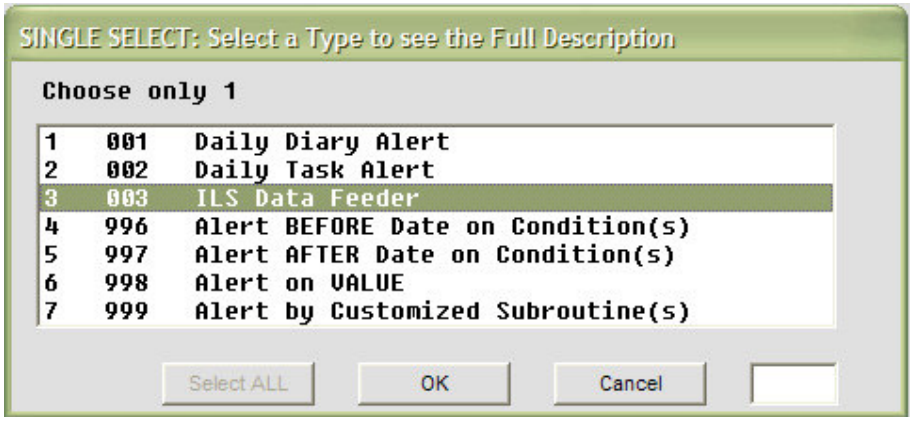
The INTERVENE option is used only if the Auto Feeder reports (in the Event Log) that it cannot proceed by itself for some reason, and will offer RESET option(s). The EXECUTE function can be used to force the Auto Feeder to execute as in NOW. Normally, the Feeder operates by itself, and checks the Home Directory for new imports at regular intervals.

The only remaining step in ensuring that the Feeder will operate by itself is to 'activate' it. This is done in INFOLAB Alerter, the system that performs automatic work of various kinds. (How to use Alerter is described in the Alerter User Manuals.)

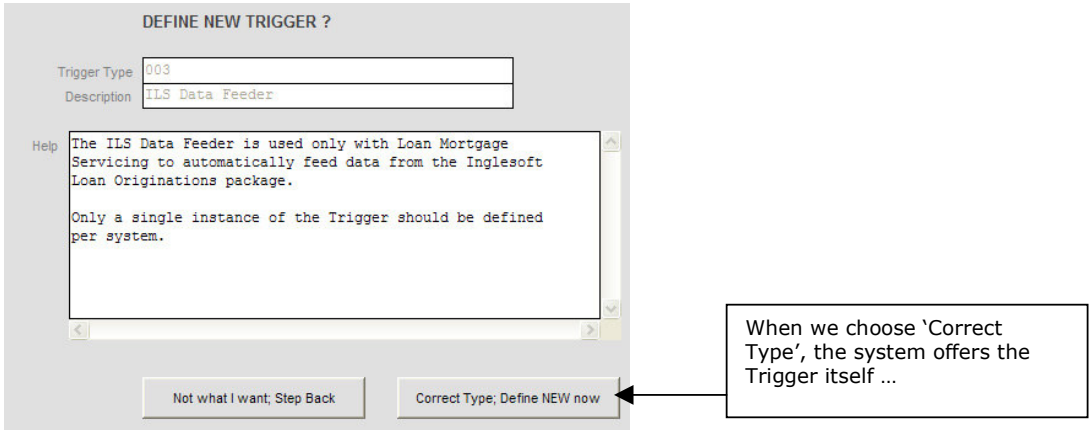
From the Alerter Wizard, choose 'User Options' then choose 'My Triggers' –

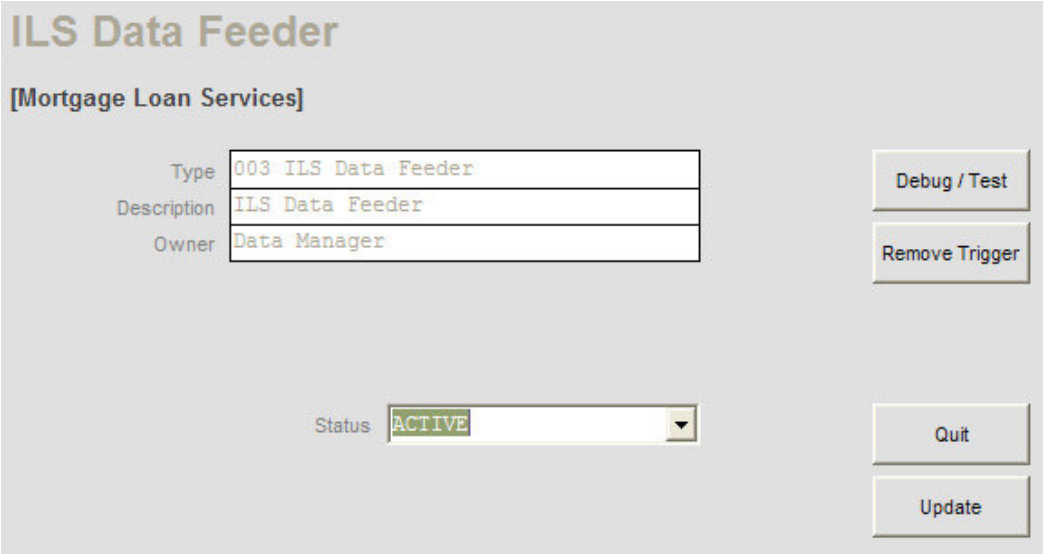


And then look for Trigger Type '003' –



Once we select this Trigger Type, it offers another screen for us to verify whether it is the correct Trigger we wish to add to our list –

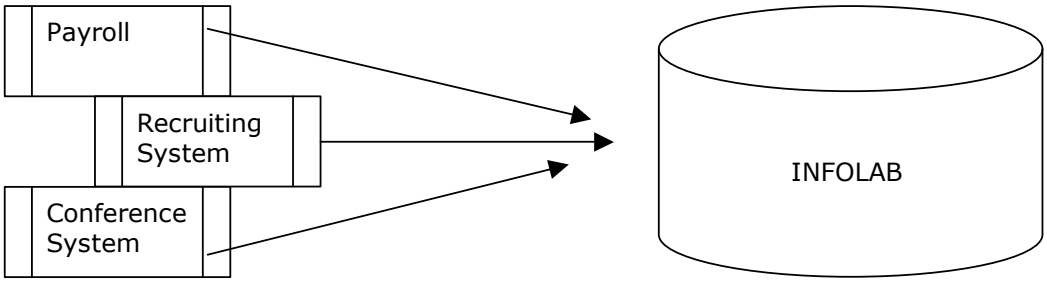




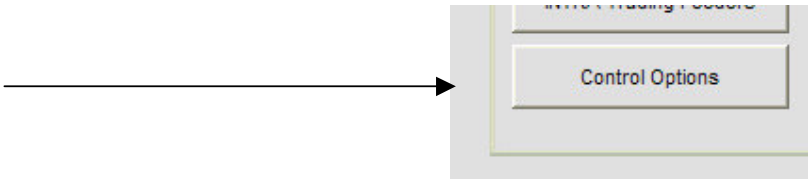
All that is required is to choose an ACTIVE status, and then choose UPDATE. Please note that it is only necessary to have ONE such Trigger active on the system for the ILS Auto Feeder to operate automatically, and of course the Feeder will ONLY operate effectively provided that the Alerter Monitor is ACTIVE (see the Alerter Operations User Manual).

JOURNAL FEEDERS

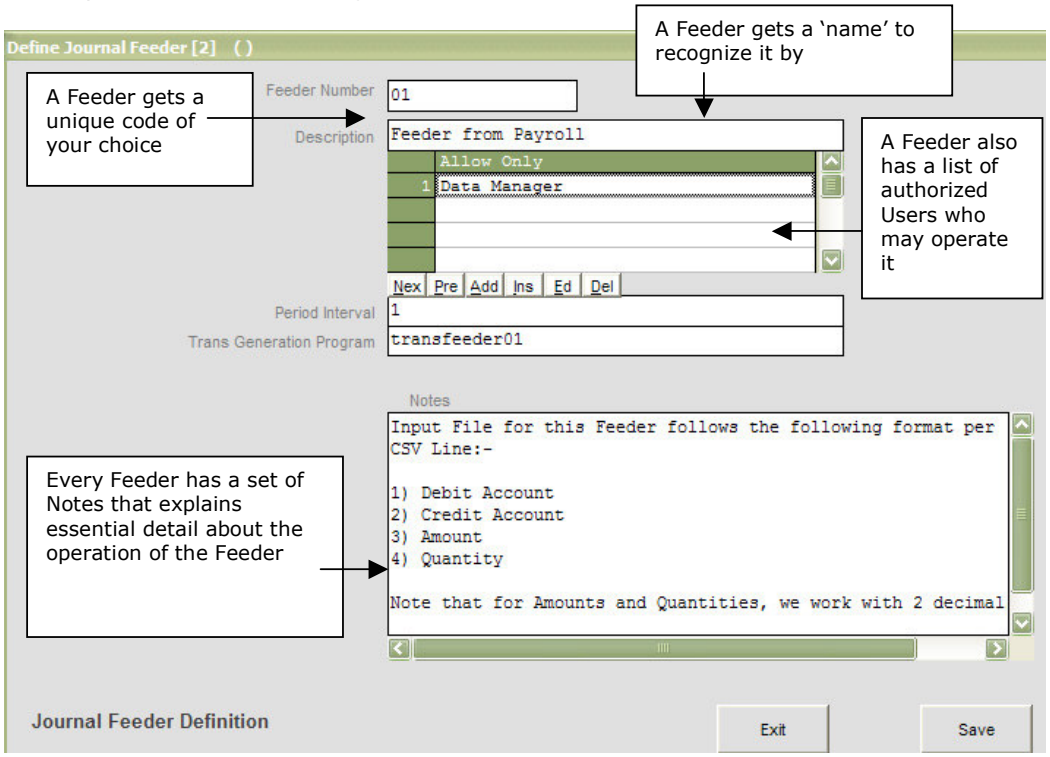
The Purpose of a Journal Feeder is to process imported data, convert it to acceptable Journal format, and post these in INFOLAB. There are certain minimum criteria required in order to construct INFOLAB Journals, but different systems may 'feed' different pieces of information, and then pass control to the specific Feeder program to provide other essential pieces of information before the Journals are constructed. Therefore, although all Journal Feeders may have the same end objective, they can use different programs and different logic to achieve that objective. As such, it is possible and even easy to create automatic INFOLAB Journals from other systems outside your INFOLAB, for example, from your Payroll system and / or other systems in use outside INFOLAB, or even originating from other Business Units in the Enterprise.



It is clear then that any number of different Journal Feeders with different processing Logic may be in use. Any Journal Feeder requires a Feeder Definition record, and this is accessed by choosing CONTROL OPTIONS –



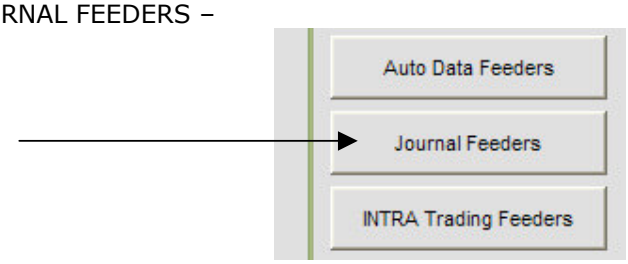
which produces the following screen form: -



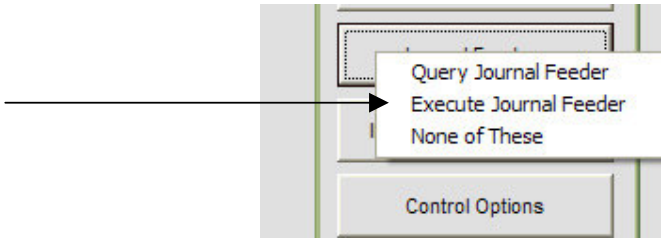
In the clip shown above, we observe also a 'Period Interval' and a 'Trans Generation Program'. The Period Interval may be set to zero to allow multiple feeds during a Financial Period, or to 1 to allow only a single feed per period. An example like a Payroll Feed is typically a case where only 1 feed per period may be set, to prevent the Payroll Journals from being imported and posted twice, but there may also be justification for allowing multiple feeds (it all depends on your requirements).

The 'Trans Generation Program' is usually set up by a Developer, since it contains program logic to deal with the specific feeder data. (This is described in the sample program 'transfeeder01' which you will find in file 'client.bp'.) Once a feeder program exists, and that operates on a data feed set, i.e. an example where imported data is in the format of passing Debit Account, Credit Account, Amount, in this sequence, per line, then the same feeder program may be used to establish other feeders that will deal with the same kind of data. In other words, a Developer is only required to 'develop' the logic when new data formats are introduced, and not for every Feeder that you wish to define.

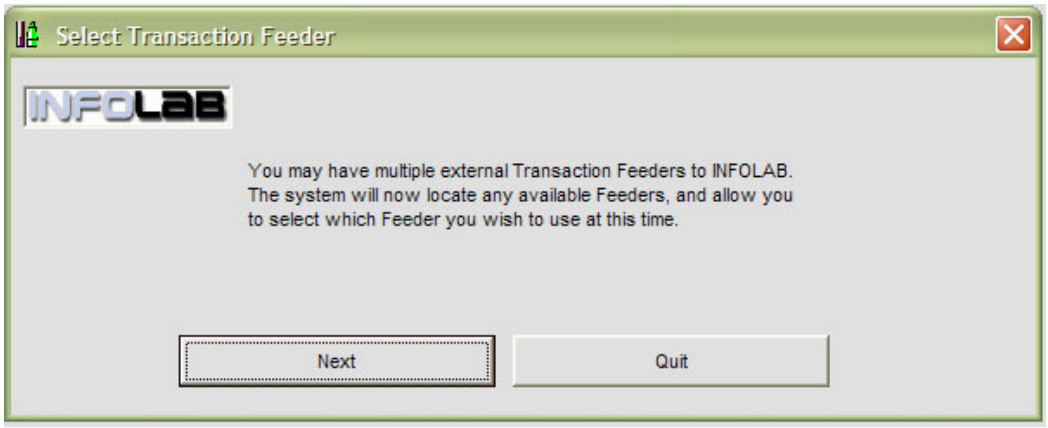
Once a Journal Feeder is defined, it may be used, and it is accessed by choosing JOURNAL FEEDERS –



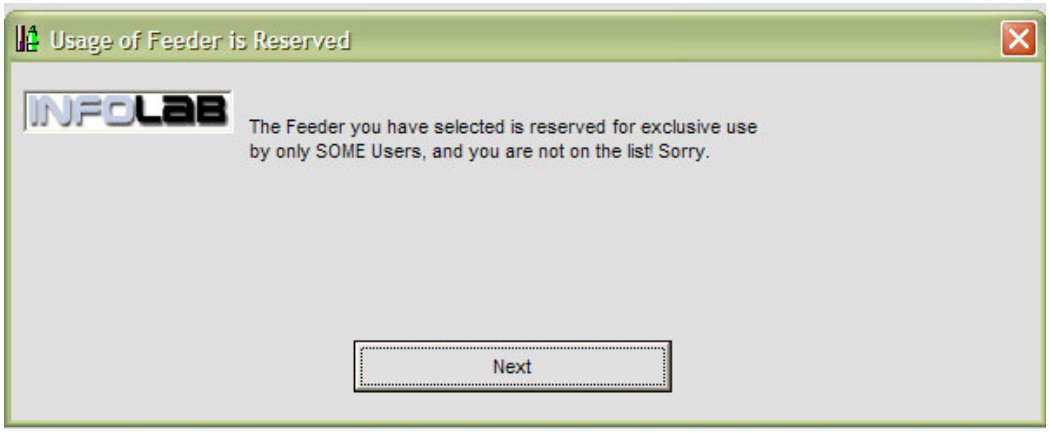
This choice opens a Wizard which will take you through the steps of operating the Feeder.



The Query option will show the same screen form as for defining the Feeder Record, but in 'look only' mode. We now choose the EXECUTE option.

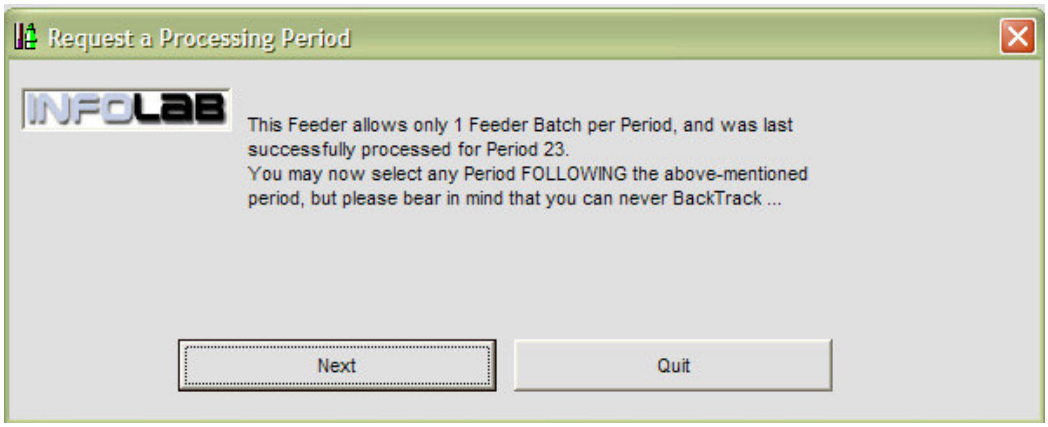


The system advises that there may be multiple Journal Feeders available. After offering the list of available Feeders, and after we have exercised our choice, the following happens.



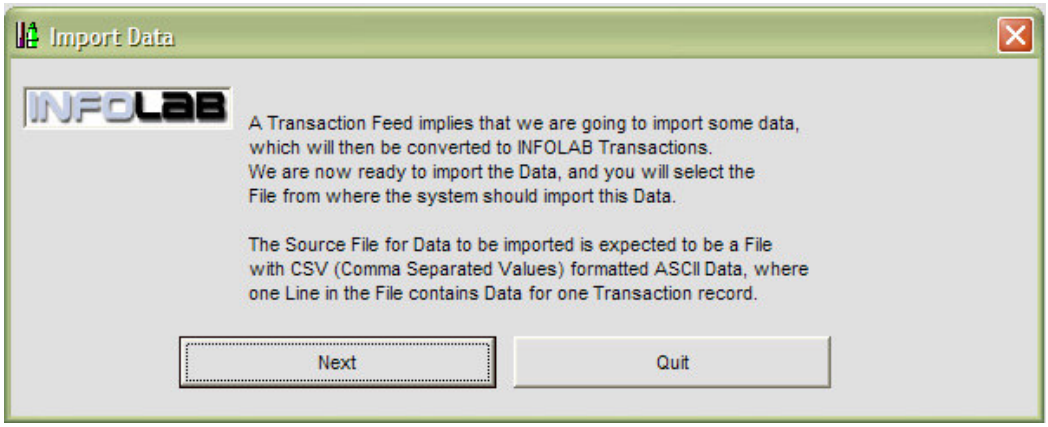
Oops, we selected a Feeder option that we are not allowed to execute. Let's revert to a valid selection …

We now select the Payroll Feeder …

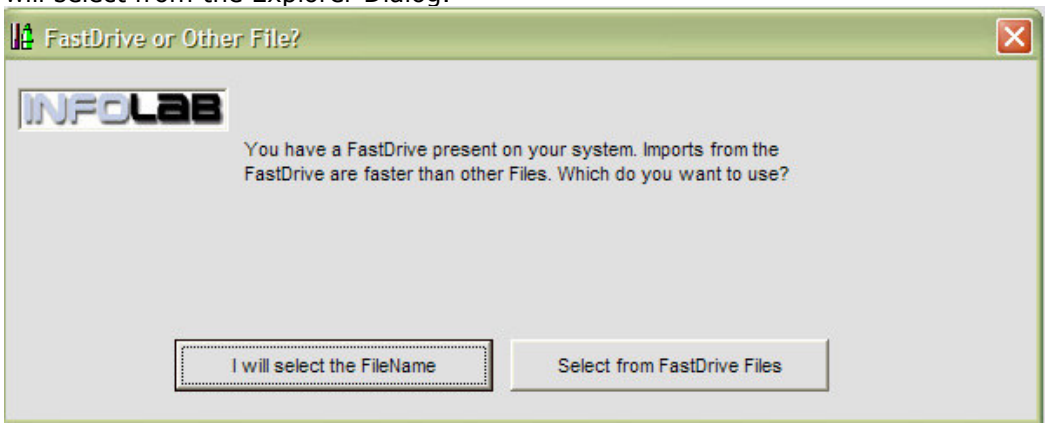


In this case, only 1 feed per Financial Period is allowed, and INFOLAB informs us that we will select a Period for execution …

After selecting the required Period, the following happens …

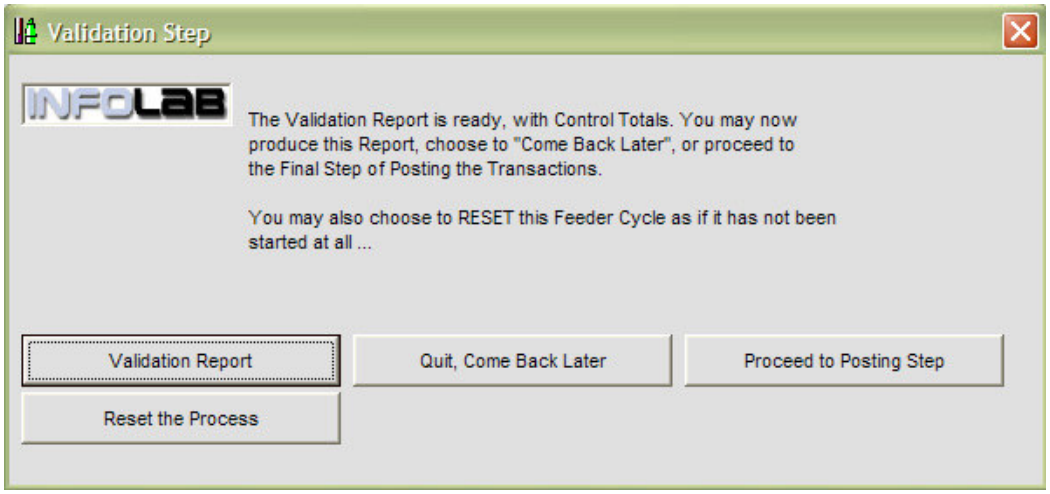


The system now explains that we are about to import Data from a File that we will select from the Explorer Dialog.

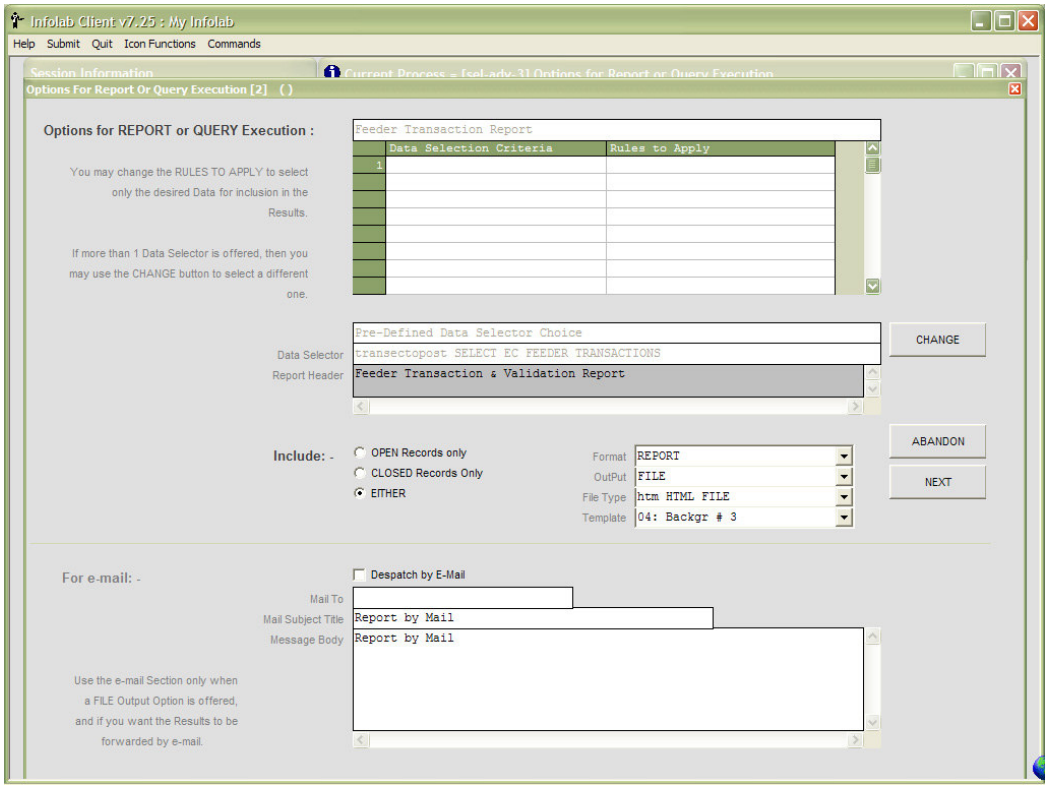


Whenever we import Data into INFOLAB and our main control Record indicates that a 'Fast Drive' has been specified, we have a choice whether to select the File from the Fast Drive contents or elsewhere. The 'Fast Drive' is nothing other than a normal Windows Folder, but one from where INFOLAB can map data much faster than otherwise. You cannot go wrong by trying the Fast Drive, because INFOLAB will offer you the list of contents, but if you do not know which Folder this is in order to place your Import Files there, then please ask your Systems Administrator. If you know your File is not in the Fast Drive, then choose "I Will Select the FileName".

Once the File is selected, the system imports it automatically, and offers the next dialog.



Quite easy to understand. If we wish to continue with this Feeder sequence, we choose the Validation Report.



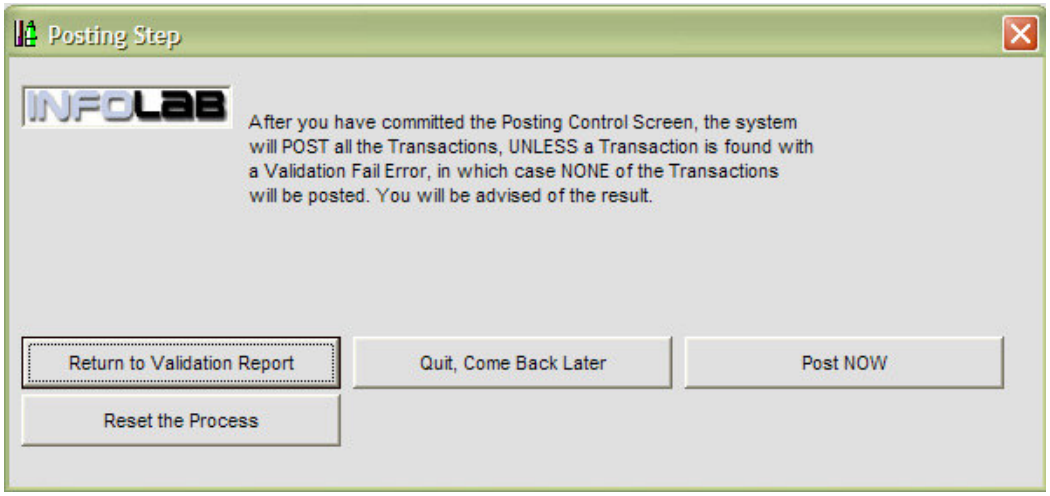
The above screen form is a standard step for report execution in INFOLAB (this is described in the User Manual that explains how to use your INFOLAB Client Software). We may accept the defaults as provided, or choose a different output format, etc. When we choose NEXT, the Report is executed.

```
21:02:00 28 Feb 2004 page    1

 Feeder Transaction & Validation Report

transectopost.. Account to Debit. Account to Credit. Amount..... Quantity..... Error Code.

000001          10101052          10201052                12.56             1.00
000002          10201052          10101052                50.00            12.00
***                                                       62.56            13.00
```

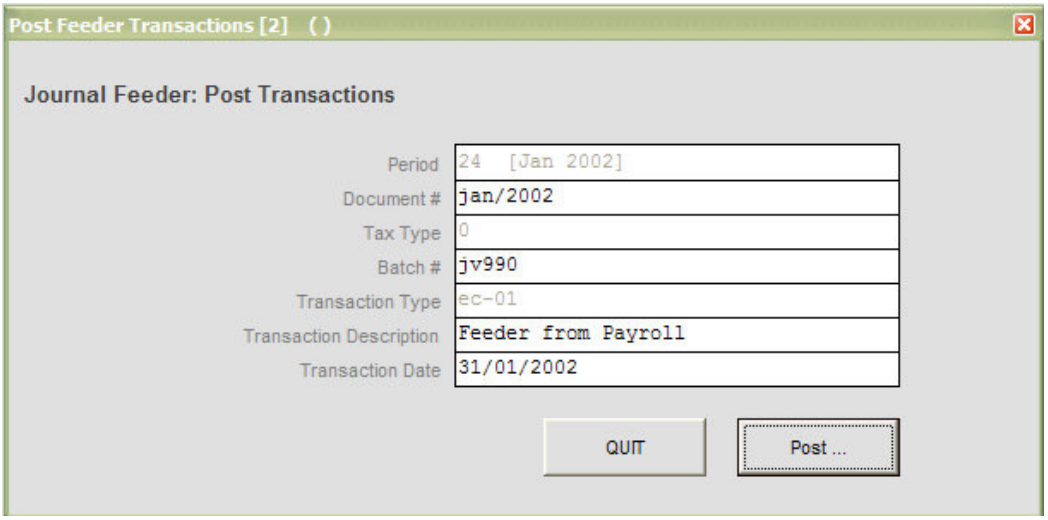
The Error Code Column should be free of any reported errors.

The Validation Report is an essential step, and we should not proceed unless we are able to produce a Report with NO error codes listed. If we post a 'feed' with Errors, then all the Journals will be rejected by INFOLAB. Therefore, if any errors are present, the Data should be corrected and the sequence re-started.

If the Validation Report is free of any errors, then we may choose PROCEED TO POSTING STEP (shown above), and the following happens …

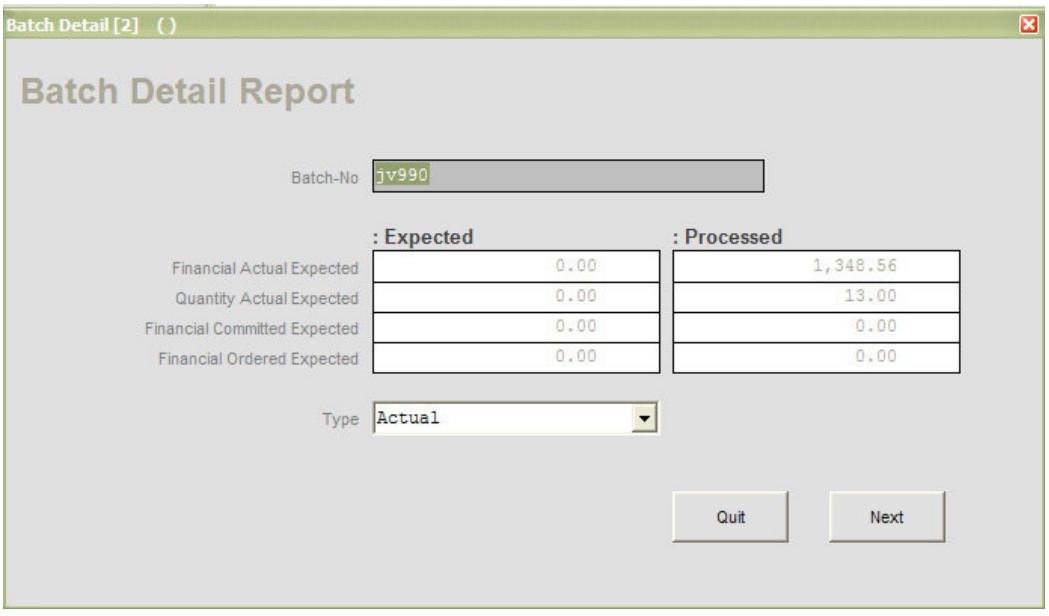


So it is fairly obvious that we should want to choose POST NOW –



The Fields that are dimmed in the picture shown above, are set by the system. The other Fields (some are defaulted), we may specify, e.g. Document Number, Batch, Transaction Description and Date. Then we choose POST –

The system automatically posts all the Journals included in this Feed, and advises accordingly. The only action remaining for this Feed is that we might like to produce a Batch Detail Report of what was posted. Remember the Main Wizard in the beginning of this Document? It includes an option to produce a Batch Detail Report, and we should use the Batch Number as specified in the screen form shown above (when posting).



The Batch Report is produced when we choose NEXT. And that concludes our explanation of how to operate a Journal Feeder.

INTRA TRADING FEEDERS

The singular purpose of Intra Trading Feeders is to close off Balances on specified Accounts in one INFOLAB, and post each as a Journal to a defined Account in the next INFOLAB. Bearing in mind that when multiple INFOLAB systems exist on the same Server, they can directly update Accounts in other INFOLABS on that Server, without any need for Intra Feeding. Also, if the need is simply to export Balances, this can be done with Data Queries, which can be translated into Journals to be processed by Journal Feeders if required. Therefore, Intra Feeders are more commonly used to transfer Account Balances (and close off these Balances in the originating INFOLAB), where the INFOLAB systems that form part of the exchange are not operating on the same Server. An example of this setup could be a Company with different Business Units, operating off different systems, but where a need exists to 'pull in' parts of the Accounts of one system into the next, possibly on a monthly (or other) basis.

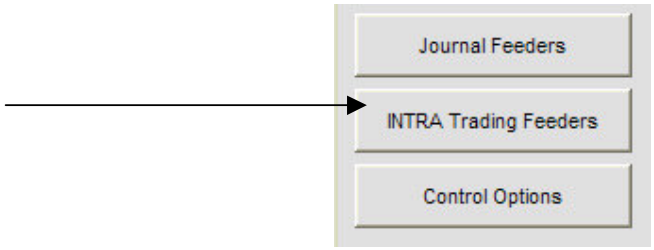
What does all of this mean? Well, let us consider a Corporate HQ that 'owns' a set of AR / Debtor Accounts. This HQ has a Property Management Division that operates an 'independent' INFOLAB in another location (other than HQ). The Property Management Division produces Invoices to some of the same AR / Debtor Accounts that are present in HQ INFOLAB. At the end of each Month, the Property Division closes off the Balances on it's AR / Debtor Accounts, and passes them along to HQ for inclusion in the Statements produced and forwarded to the Customers. HQ then 'owns' the outstanding Amounts, and collects form the Customer base the settling of these Invoices (included in the Balances that are transferred). At the same time, when Property passes the Balances to HQ and HQ imports them, INFOLAB automatically manages Trading or Loan Accounts between the 2 INFOLAB systems. So let us examine a small example of what this is about.

Let us suggest that at the start of the month that we are examining, the 2 INFOLAB systems 'owe' each other Zero.

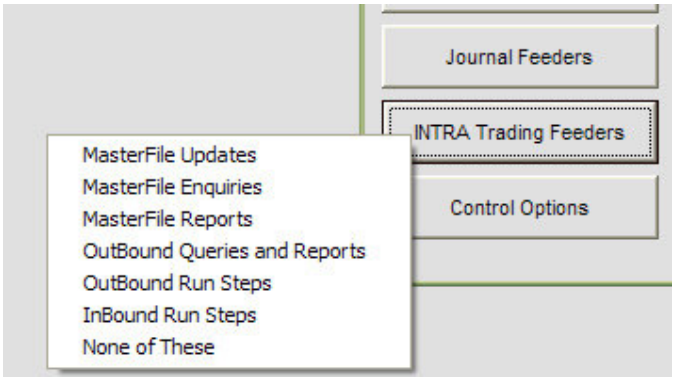
1) Property raises an Invoice to AR Account 0000012, in the amount of $2500-00.
2) Property raises an Invoice to AR Account 2300011, in the amount of $523-00.
3) At month end, by means of an Intra Feeder cycle, Property clears out these 2 AR Accounts, totaling $3023-00, and each of these Accounts now reflect a Zero Balance, i.e. as if they have been settled.
4) At Property, the Loan Account with HQ is Debited with $3023-00, and this amount is owed to Property by HQ.
5) At HQ, the Intra Feeder cycle imports Journals to the Value of $3023-00, which is credited to the Property Loan Account, now reflecting a Credit Balance (i.e. owing to) of $3023-00.
6) At the same time, 2 AR Debtor Accounts at HQ (possibly with different Account Numbers), are debited (charged), with $2500-00 and $523-00, respectively.
7) Therefore, HQ now 'owes' Property the amount of $3023-00, which HQ will collect from AR Debtors.

This is a simple example, but the Intra Feeders, once set up, can automatically 'manage' any number of Ledger, AP / Creditor and AR / Debtor exchanges between any number of Business Units, with minimum fuss. The Loan Accounts employed are auto managed and reconciled by INFOLAB without any effort or intervention by Users (Accounts Staff).

It is time to see how setting up and using the Intra Trading Feeders can achieve all of this.

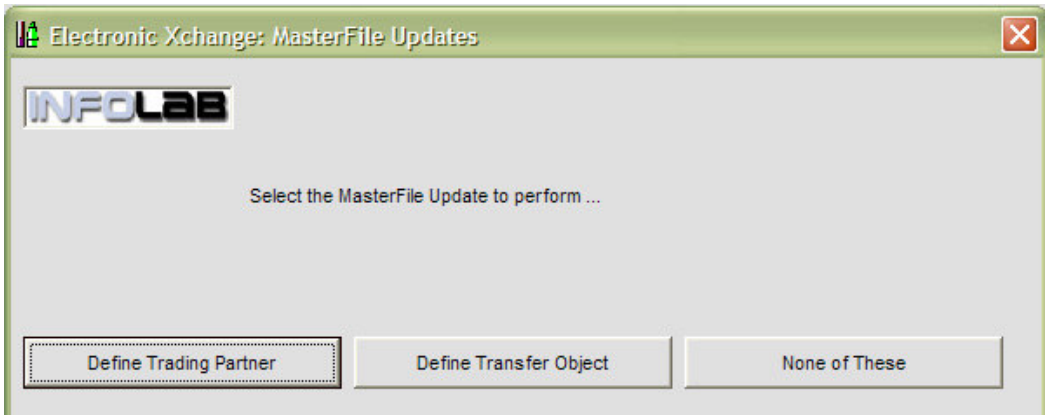


This selection leads to further choices –

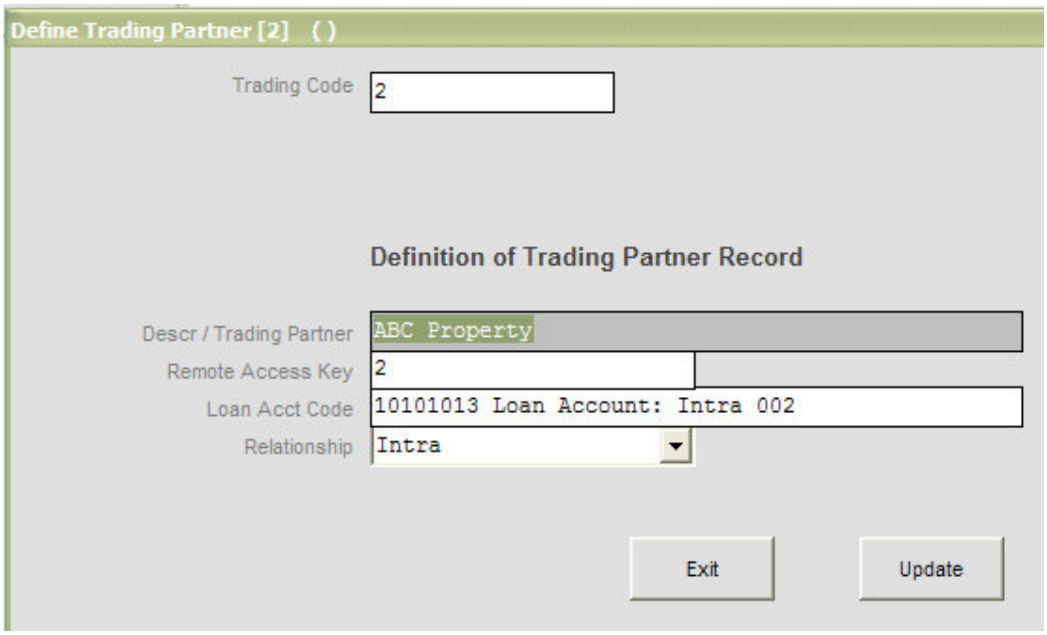


The Intra Feeder options include more options for MasterFile Updates, Enquiries and Reports, and there are OutBound Queries and Reports, and, as may be expected, there are Sequential Steps for OutBound and InBound Runs.
Look us have a look at each of this, in turn.

Master File Updates

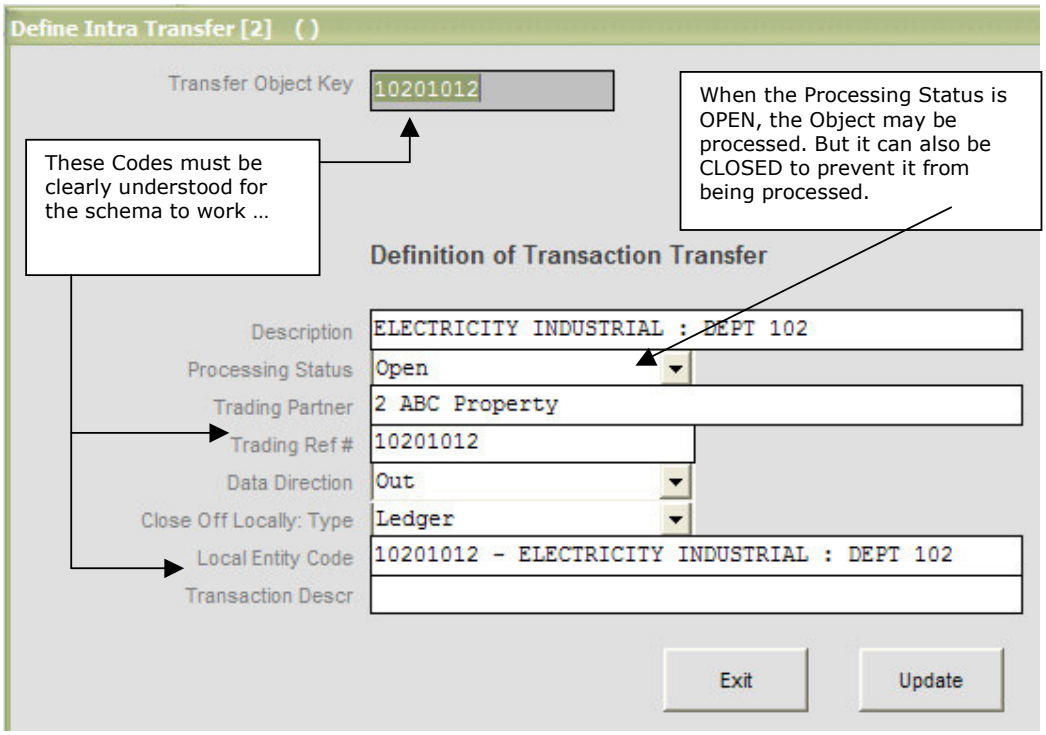


There is a requirement to define Trading Partners and to define Transfer Objects. The Trading Partner records identify the different Business Units that exchange Intra Feeds, and the Transfer Objects described the Account References on the various sides. Here is the definition of a Trading Partner –



The Trading Code is of your own design, but for the sake of Data Exports to Excel or other, it is recommended that the code does not start with leading zeros. The Remotes Access Key refers to the Code used by the Trading Business Unit when it references HQ. These 2 codes can be the same, or different. The Loan Account Code references the Loan Account in the Accounts of HQ, which at any point in time reflects the Trading balance with the other Business Unit, e.g. Property Division. On the Property side, there is a similar record that describes HQ as a Trading Partner, and where the Trading Code is the same as what is indicated above as 'Remote Access Key'.

And now for a look at the Transfer Objects –



A Transfer Object describes an Account that may 'feed'. Such Accounts may be of types Ledger, AP / Creditors and AR/ Debtors. Note that there is always an indicated 'direction', i.e. OUT to the Trading Partner or IN from the Trading Partner. If the same Account has to do both, you need to define 2 Transfer Objects to achieve it. Which leads us to one of the reasons why the Transfer Object Key does NOT have to be the same as the Account Code, nor has to be the same as the Code with which the Trading Partner refers to the Transfer Object.

'Transfer Object Key' is the Key for the Object in the local INFOLAB system, in this case HQ.

'Trading Ref #' is what the Trading Partner uses as 'Transfer Object Key' in their INFOLAB.

'Local Entity Code' is the Account Code in the local INFOLAB, and immediately above this Field is an indicator that determines which type of Account it is, e.g. Ledger, AP, AR.

'Transaction Description' is optional. When it is indicated, it is used on the Journals create by the Object. When it is not indicated, then the global Transaction Description for the 'run' is used.

Not so daunting after all, is it?

Master File Enquiries

The Enquiries are exactly the same as the Updates, except that they are offered with 'look only' access, i.e. Users who are restricted to the Enquiry Options may only look at the Definitions, but are not allowed to change them.

Master File Reports
The Master File Reports offer options to list the Trading Partners, or to list the Transfer Objects (and of course you may restrict the listing to selected criteria).

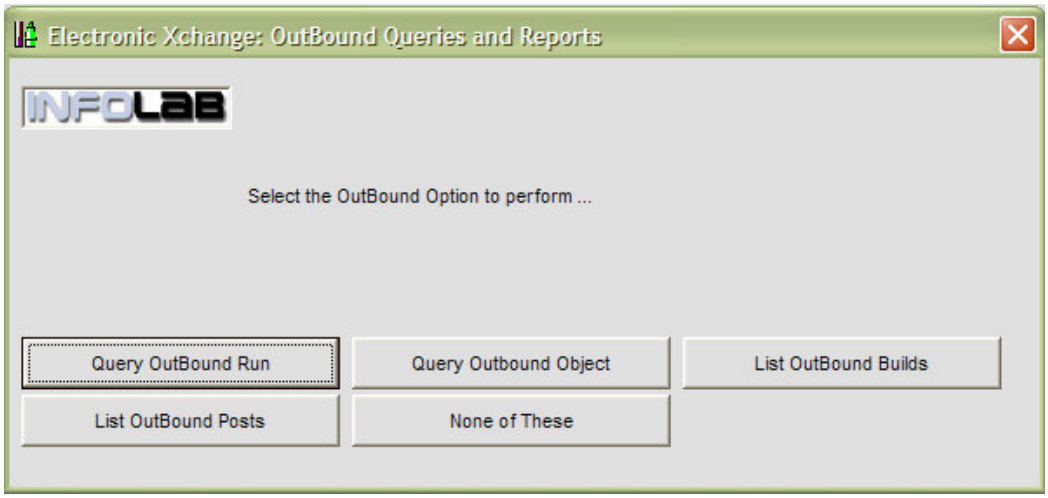
```
Electronic Trading Partners

Key..... Descr / Trading Partner..... Loan Acct Code.....

1              ABC Corporate HQ              10101012 Loan Account: Intra 001
2              ABC Property                  10101013 Loan Account: Intra 002
3              ABC Private School            10101014 Loan Account: Intra 003
```

Above we have Trading Partners, and below we have Transfer Objects –

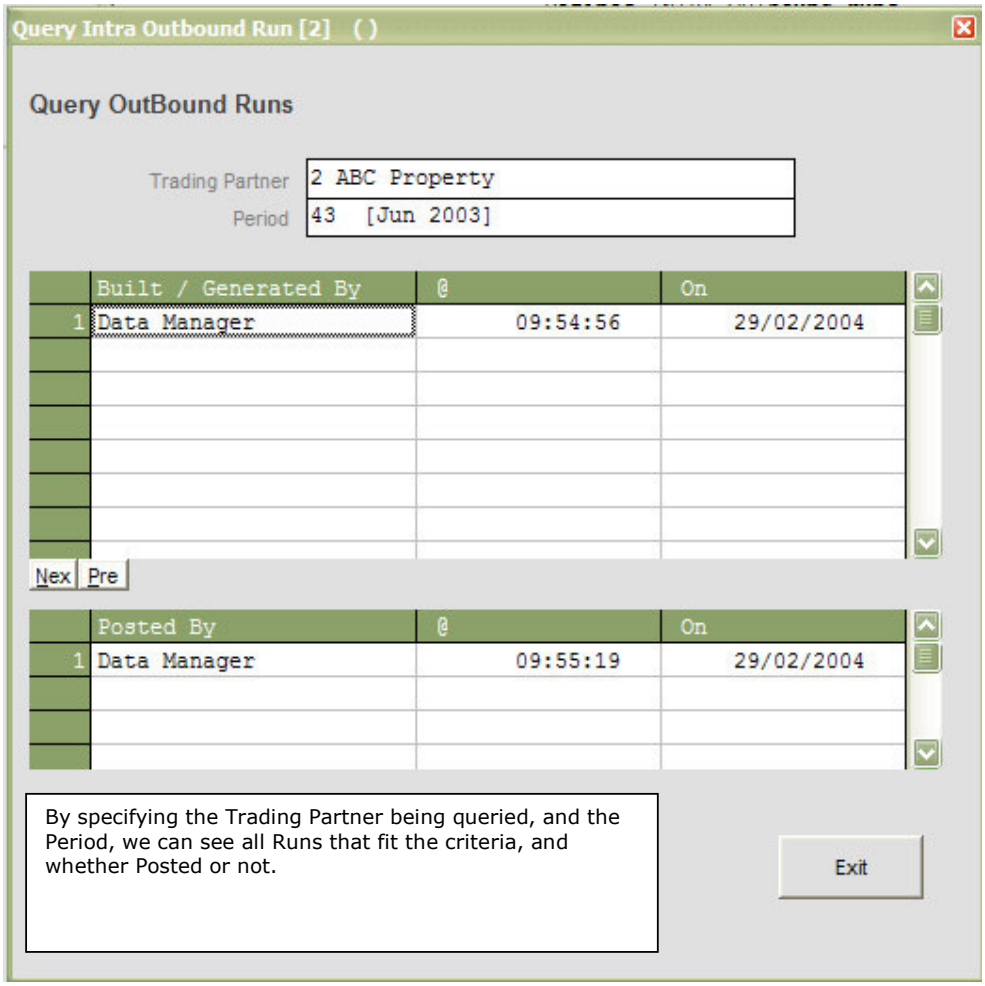
```
Electronic Transfers

Key..... Description..... Partner. Type... Acct Code..... In/Out OC..... Ref #...

10201012       ELECTRICITY INDUSTRIAL : DEPT 102   2       1       10201012       Out     Open    10201012
10201013       SUNDRY STORES & CHARGES : DEPT 102  2       1       10201013       Out     Open    10201013
10201014       DONATIONS : DEPT 102                3       1       10201014       In      Open    10201014
10201015       MEDICAL EXPENSES : DEPT 102         3       1       10201015       In      Open    10201015
```

OutBound Queries and Reports

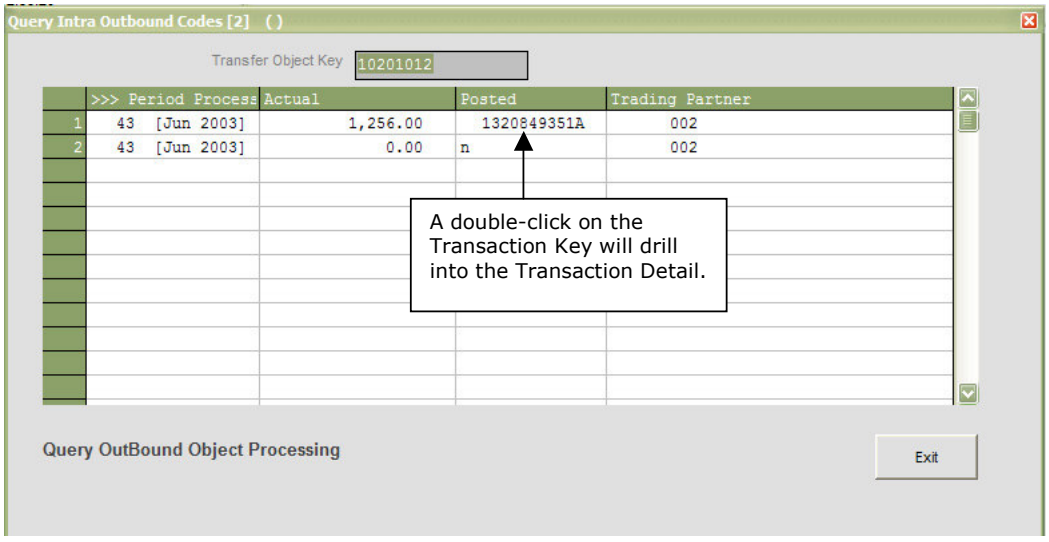
The OutBound Queries and Reports include the following –



Query Outbound Run is an option to view the status of the OutBound Runs for a selected Partner within a selected Period.



We can likewise Query any OutBound Transfer Object –



In the example above, we are querying an OutBound Transfer Object, which will list all Periods for which it was ever Processed and Posted. In the 'Posted'

Column, if the Transaction was posted, we may double-click on the Transaction Key to drill into the Detail for that Transaction.

The OutBound Builds Report will detail Builds for the specified Criteria, e.g. Partner and Period. Bear in mind that a Run may be Built and re-Built several times until we are satisfied with the results, before proceeding with Posting, if so required.

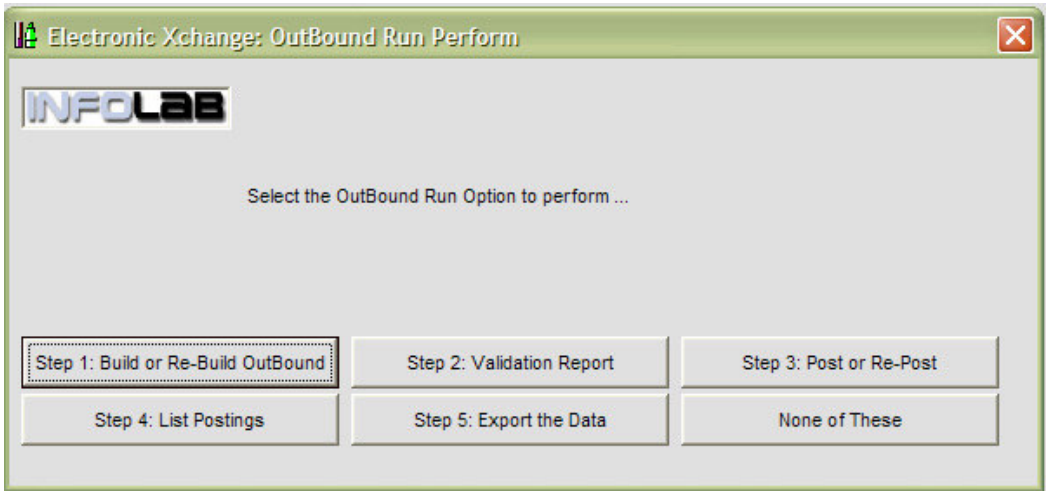
```
13:00:48 29 Feb 2004 page    1

 Run Build Report, per selection.

Period.............. Trading Partner..................... Built / Generated by  @ ............  on ...........
                                                                                 *              *
43  [Jun 2003]       2 ABC Property                      Data Manager              12:44:46      28/02/2004
                                                         Data Manager              12:52:46      28/02/2004
                                                         Data Manager              12:54:54      28/02/2004
                                                         Data Manager              12:56:44      28/02/2004
                                                         Data Manager              13:11:44      28/02/2004
                                                         Data Manager              13:37:31      28/02/2004
                                                         Data Manager              14:13:25      28/02/2004
43  [Jun 2003]       2 ABC Property                      Data Manager              09:54:56      29/02/2004
```

Similarly, the OutBound Posts Report can be produced for selected Partners and Periods, as specified.

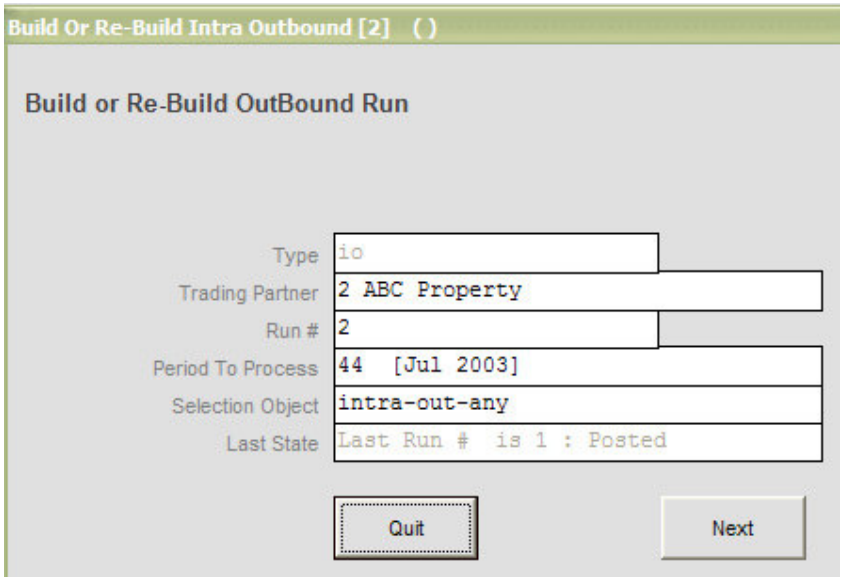
```
13:03:54 29 Feb 2004 page    1

 Posting Report, per selection.

Period.............. Trading Partner..................... Posted By...........  @ ............  on ...........
                                                                                 *              *
43  [Jun 2003]       2 ABC Property                      Data Manager              13:42:31      28/02/2004
43  [Jun 2003]       2 ABC Property                      Data Manager              09:55:19      29/02/2004
```

OutBound Run Steps

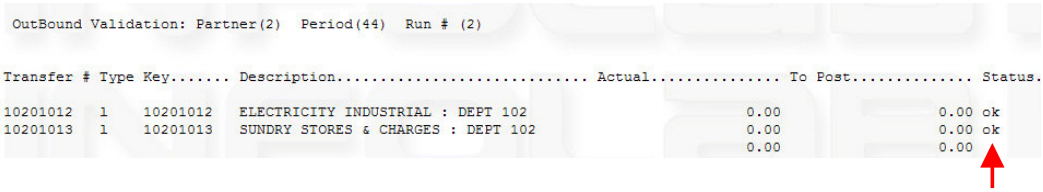
An OutBound Run is performed in a set of 5 steps –



Step 1 entails the generation of the Transfers, which is called a 'Build'. We may specify criteria to determine what will be included in the Build, i.e. by Partner, and with an appropriate Data Selector (which determines which Transfer Objects will be selected for inclusion), and we also choose the Period for which the system will determine what the closing balances on these Accounts are.



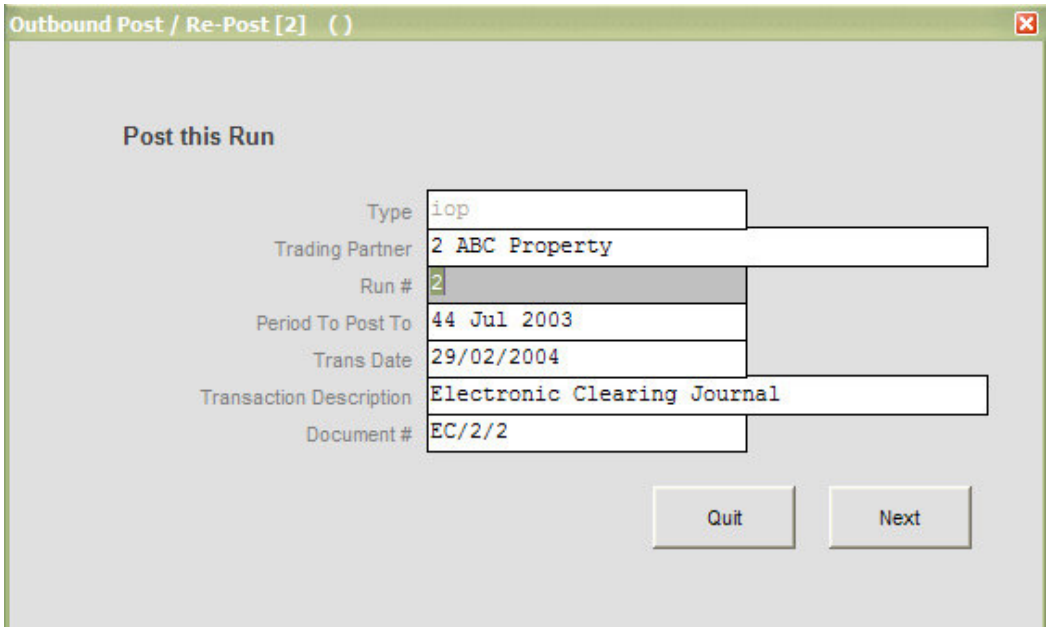
When we choose NEXT, the system generates the BUILD. Our next step is to produce a Validation Report to check the Results (Step 2 as shown above).



```
OutBound Validation: Partner(2)  Period(44)  Run # (2)

Transfer # Type Key..... Description..... Actual..... To Post..... Status.

10201012   1    10201012   ELECTRICITY INDUSTRIAL : DEPT 102          0.00          0.00 ok
10201013   1    10201013   SUNDRY STORES & CHARGES : DEPT 102         0.00          0.00 ok
                                                                     0.00          0.00
```

It is important to verify that the system reports no errors, i.e. Accounts that are Closed, Not Found, etc.

If we have problems, we can correct them and re-generate the same Build again. Once the Validation Reports is clear if any reported problems, it is time to POST (Step 3).



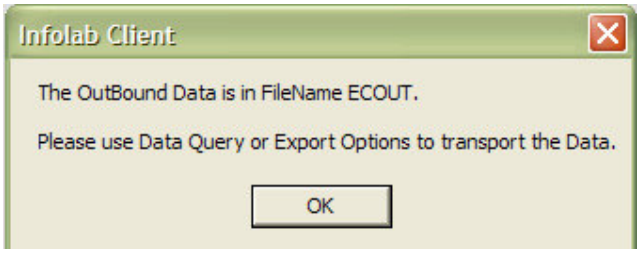
We are allowed to change the Document Number, Transaction Description and Transaction Date if any of the Defaulted Values do not suit. The system will then perform the Posting automatically, and once again will report if any problems occur, in which case these can be attended to and a Re-Post performed (the system controls the sequence adequately, and will not post the same Closing Balance twice).

Step 4 is a 'Posting Report' to verify what the system has posted, i.e. what will be exported to the Trading Partner.

```
OutBound Posting: Partner(2)  Run # (1)

Transfer # Type Key..... Description.............................. Actual.............

10201012   1    10201012   ELECTRICITY INDUSTRIAL : DEPT 102                  1,256.00
10201013   1    10201013   SUNDRY STORES & CHARGES : DEPT 102                     15.00
                                                                              1,271.00
```

The Posting report can be done not only for the current Build, but also for any prior Build. In the example above, we show the Posting Report for Run # 1, even though in Step 3 (above) we were working with Run # 2.

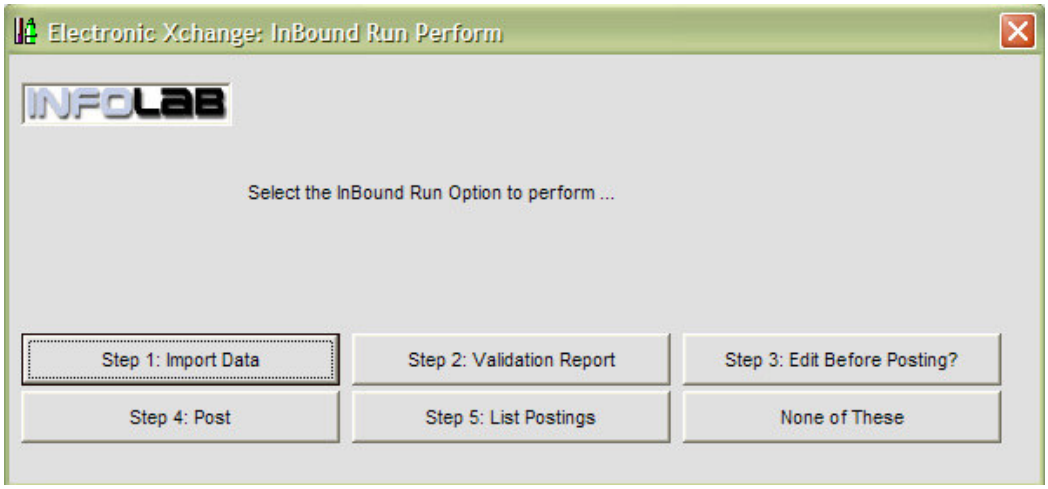
Step 5 entails the Exporting of the Data –



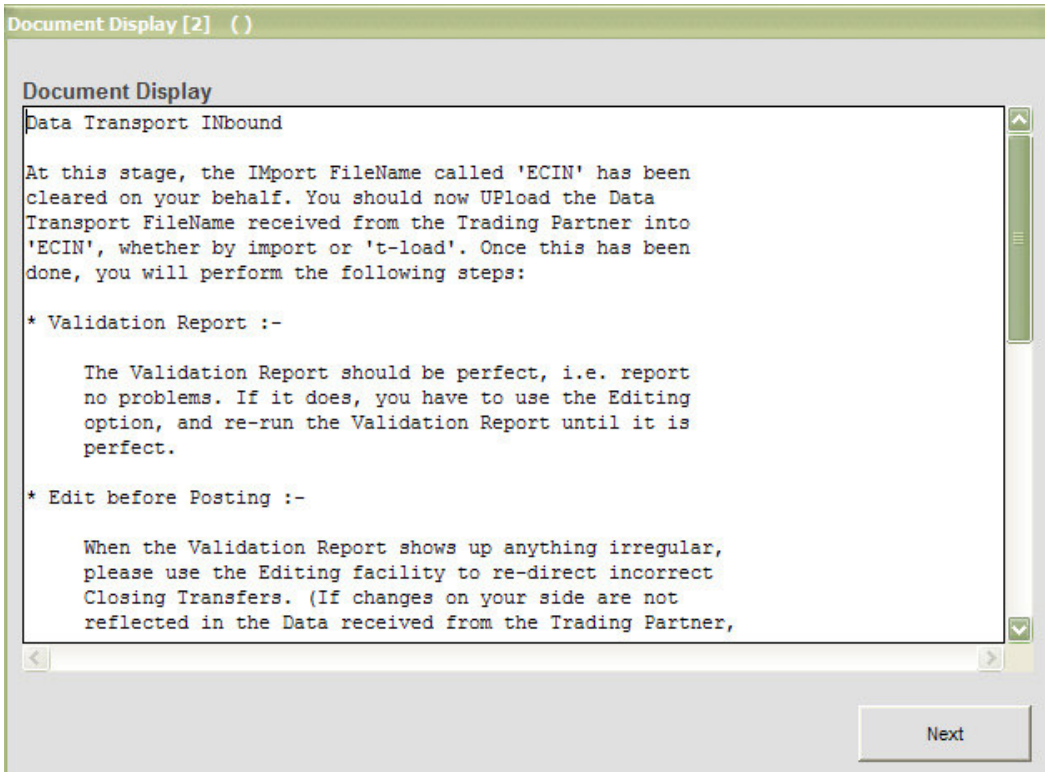
And the system merely reminds us that the Export Data resides in file ECOUT, and we may use either the provided, pre-defined Query, or another Query of our own design, to export the Data for subsequent import into the Trading Partner's INFOLAB. (Remember that a quick link to QUERY BUILDER is provided on the main Electronic Exchange Wizard).

InBound Run Steps

An InBound Run (received from a Trading Partner), is performed in 5 steps.

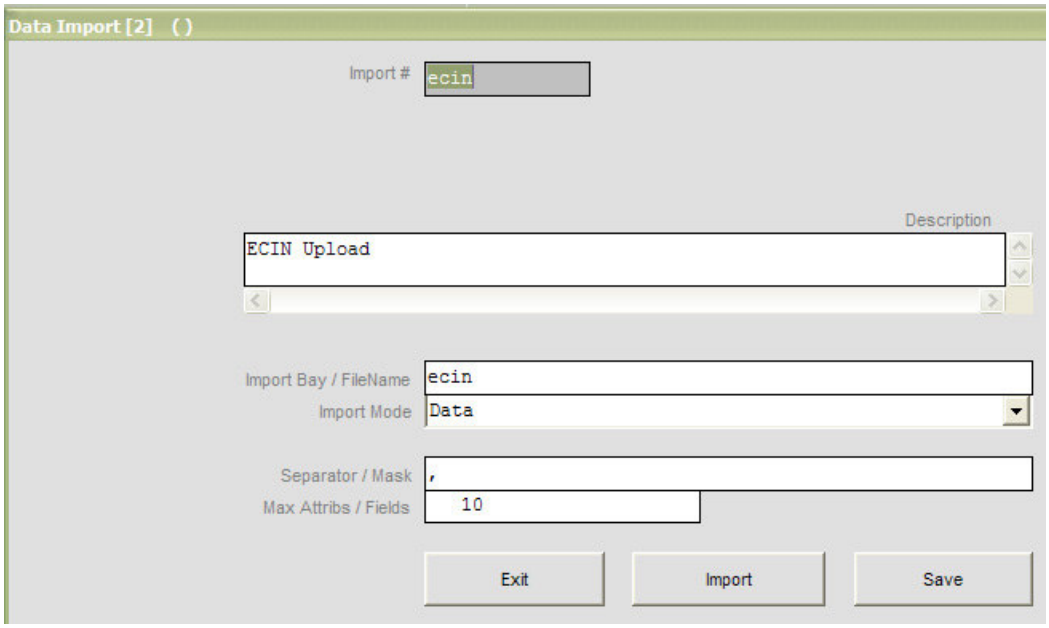


Step 1 entails the Import of the Data File received (remember the DATA IMPORT option on the main Wizard?).

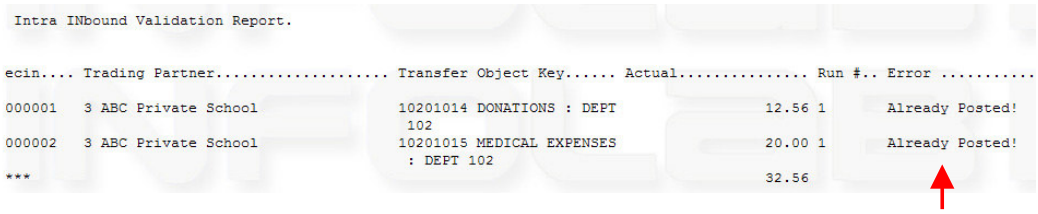


When we choose Step 1, the system offers a Document Display that explains the steps to perform during the InBound Run.

We can exit the Steps temporarily, use the DATA IMPORT option, and we can use the standard (pre-defined) Import Option for ECIN (the File we should import into) –



When we choose IMPORT, the system will offer the Dialog for us to select where to get the Data from, and, as explained before, we can select from the Fast Drive options or select a FileName from another Folder. The system then automatically imports the Data, and we follow up this Import Action by performing Step 2, which is a Validation Report the system produces based on the imported Data.

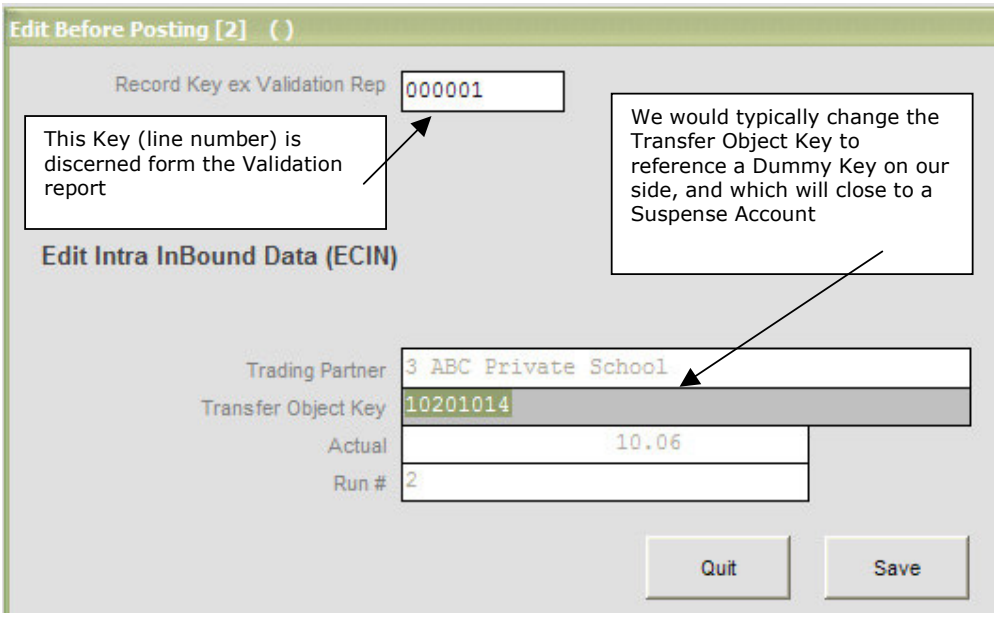


```
Intra INbound Validation Report.

ecin.... Trading Partner..................... Transfer Object Key..... Actual.............. Run #.. Error ...........

000001   3 ABC Private School                 10201014 DONATIONS : DEPT          12.56 1       Already Posted!
                                              102
000002   3 ABC Private School                 10201015 MEDICAL EXPENSES          20.00 1       Already Posted!
                                               : DEPT 102
***                                                                             32.56
```

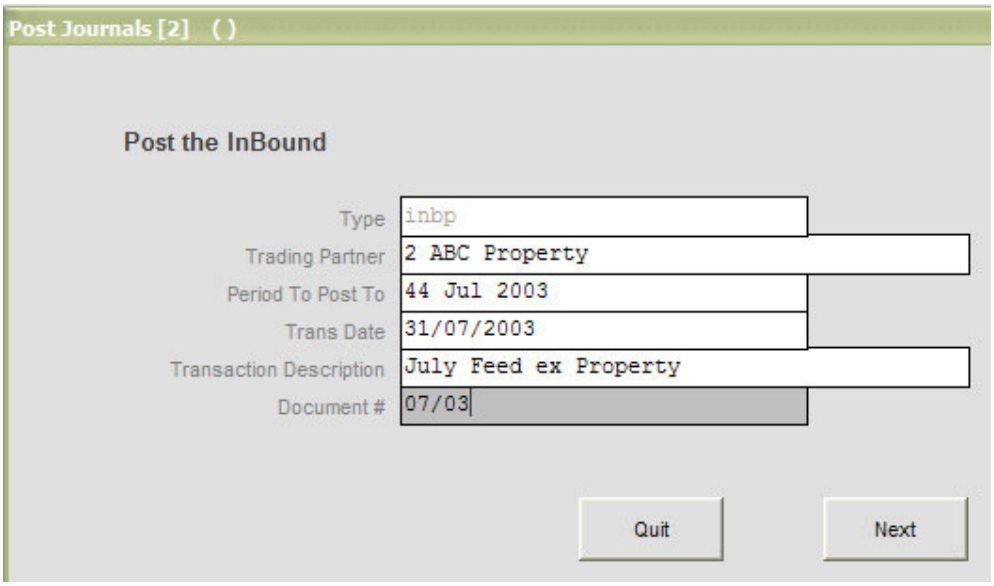
This Validation Report is crucial, and should be entirely free of reported Errors. In the example shown above, the wrong File was imported, since it is for a Run that has already been posted.

Bear in mind that INFOLAB at HQ and INFOLAB at Property auto manages Loan Accounts that reflect the Runs that pass between the two Partners. Therefore, it is essential always to Post the entire contents of a Run, and this can only be done if there are no errors reported.

Some errors that can occur, is that the Partner is referencing a Transfer Key that does not exist on HQ side, or for an Account that has since been closed. However, we CANNOT exclude these updates. Therefore, we will introduce 1 or more 'dummy' Transfer Objects, which post to a Suspense or Hold Account, and we correct any mistakes in the Feed by editing the Imported Line Objects (Step 3), and substituting our Dummy Transfer Object for lines that reference Transfer Keys that cannot be executed, because they produce errors on our side. The entries that are subsequently posted to our 'suspense' Account, are then followed up with the Trading Partner for correction in the next Run, or posted out to other Accounts in our system, which reference the Trading Partner, but we obviously do NOT post to the Loan Account itself.



We then produce the Validation Report again, and we repeat this until we have a Validation Report that is free of any errors. Then only is it time to proceed with Step 4, i.e. POSTing.



After specifying to Period to Post to, the Date to use, and the Description and Document number, we may select NEXT for automatic Posting to take place. And once this is done, only Step 5 remains, i.e. to produce the Postings Report. The grand total for the Postings should agree with the Grand Total on the Validation (Imported Data) Report. And that's all there is to it!