appx SOFTWARE

Language Translation

Version 4.0

appx SOFTWARE

TABLE OF CONTENTS
TABLE DES MATIÈRES
TAVOLA DELL'INDICE

DRAFT COPY

October 8, 1999

OVERVIEW AND REQUIREMENTS
ÜBERBLICK UND ERFORDERNISSE
VISTA GENERAL Y REQUERIMIENTOS

The *APPX Language Translator* provides the capability of presenting APPX screens to users in multiple languages. Menus, Input Processes, Inquiries, and even Queries can now be displayed to users in their native language. Just because your APPX applications were designed in English doesn't mean that your employees around the world have to struggle to decipher a language that's foreign to them. APPX Language Translator makes the translation process easy and even works in multilingual environments where some users require screens in Spanish, other users read English, and still others prefer Portuguese. With the APPX Language Translator utility, your APPX applications provide the same benefits for your business . . . anywhere in the world.

The Translator consists of two basic components:

1) *The System Administration Language Translator* contains translations for words, phrases, messages, and text that are common to all APPX applications.

2) *The Application Translator* resides within each application. This part of the Translator provides translations for items specific to that application and also provides for a means of overriding System Administration level translations for that application only.

 For example, the message "ERROR – Record Not On File!" might reside in the System Administration Translator because it is used by several—if not all—applications; while the field name "Customer Past Due Balance" would probably exist only in the Accounts Receivable application's Translator because it is unique to that application.

Each of the two translator modules contain *a Dictionary File* and a *Translation File*. The Dictionary contains all words, phrases, messages, and text for all defined languages while the Translation File contains links between dictionary entries of one language to respective entries in other languages. Additionally, the Application Translator contains an *Exception File*. Translations in the application that must override those defined at the system or application level are defined here.

NOTE: The APPX Language Translator only translates design file components at this time and does not provide for a means of translating field contents residing in APPX data files and most work fields (except for fields with *Token* or *Validation Table* values and alpha/text Work Fields). For example, if an Input Process displays the tag "Product Description" followed by the field value of "Copy Paper – Letter Size Yellow", the tag of "Product Description" would be translated but the field value would not.

Setting up the Language Translator consists of these four steps:

1) *Getting Started* by defining the Design (usually English) and Presentation Languages
2) *Defining System Level Translations* to make translations common to all applications
3) *Defining Application Level Translations* to make translations specific to a single application
4) *Selecting the presentation language* to display screens in the selected Presentation Language

The only requirement necessary to use the APPX Language Translator is reading and writing fluency in the language your APPX applications are written in <u>and</u> the language or languages the applications will be translated to. The Language Translator is part of all APPX Versions 4.0 and later so no additional software is required. In fact, if you have APPX 4.0 or higher installed on your machine, you're reading to begin making your APPX applications international in scope right now!

APPX is ready!

Whenever . . .
and *wherever* you are!

GETTING STARTED
OBTENIR ONT
EMPEZANDO

1. CREATING THE SYSTEM DESIGN FILES

➤ From the APPX Main Menu, *select* System Administration, System Setup, System Administration File Management, then Create Files. This creates the System Administration level files needed for Language Translations (DICTNARY, LANGUAGE, XLAT) if the files do not already exist.

2. DEFINING THE LANGUAGES

➤ Return to the System Administration Menu and *select* System Administration, then Language Translation. You should see the menu shown in *Figure 1*.

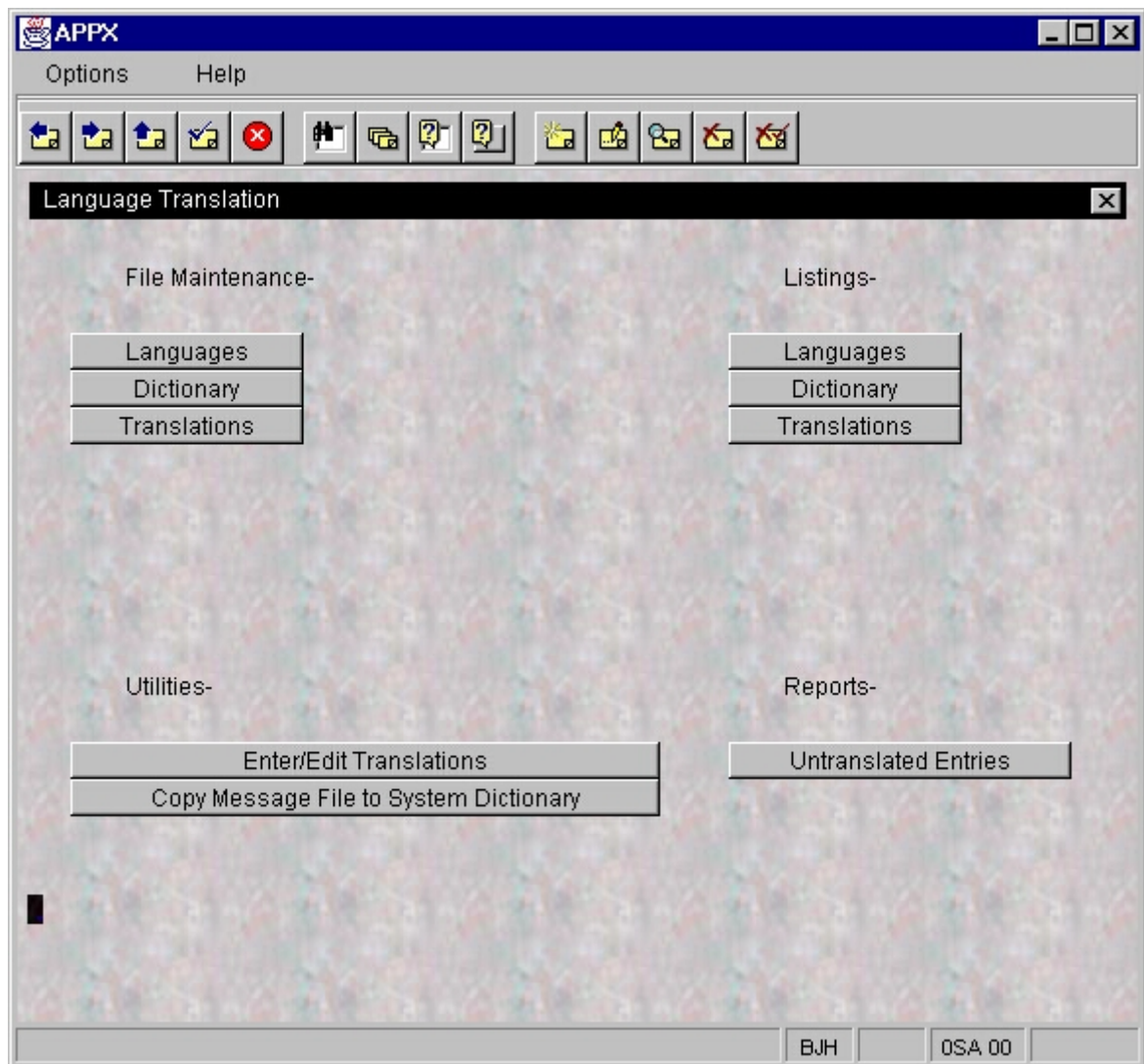*Figure 1 – The Language Translation Menu (From System Administration)*

➢ Under File Maintenance, *select* Languages.

➢ Using the Languages Input Process, *enter* a code and description for the language your APPX applications are designed in. This language will be referred to as the *Design Language* throughout this document. A list of the languages used in this document is shown in *Figure 2*.

➢ *Press* ENTER to display the rest of the Language file fields shown in *Figure 3*. Note that this example is for French.

➢ **Enter** the language's Upper Case letters and corresponding Lower Case letters (A-Z and a-z for English), then punctuation marks used as Sentence Terminators (. ! ? for English) and Clause Terminators (, ; : for English). The remaining fields (Date Mask, Currency Symbol and Position, and Decimal Notation and Grouping) are optional and are needed only if different from the values defined in the System Administration's System Parameters file.

➢ *Press* ENTER to complete the addition of the Design Language. **Repeat** the preceeding steps to add additional languages. These other languages are referred to as *Presentation Languages* throughout this document.

With the Design and Presentation Languages defined, you are ready to begin defining the translations that are common to all applications within the system.   Continue with Section 3, *The System Translator*, for further instructions.
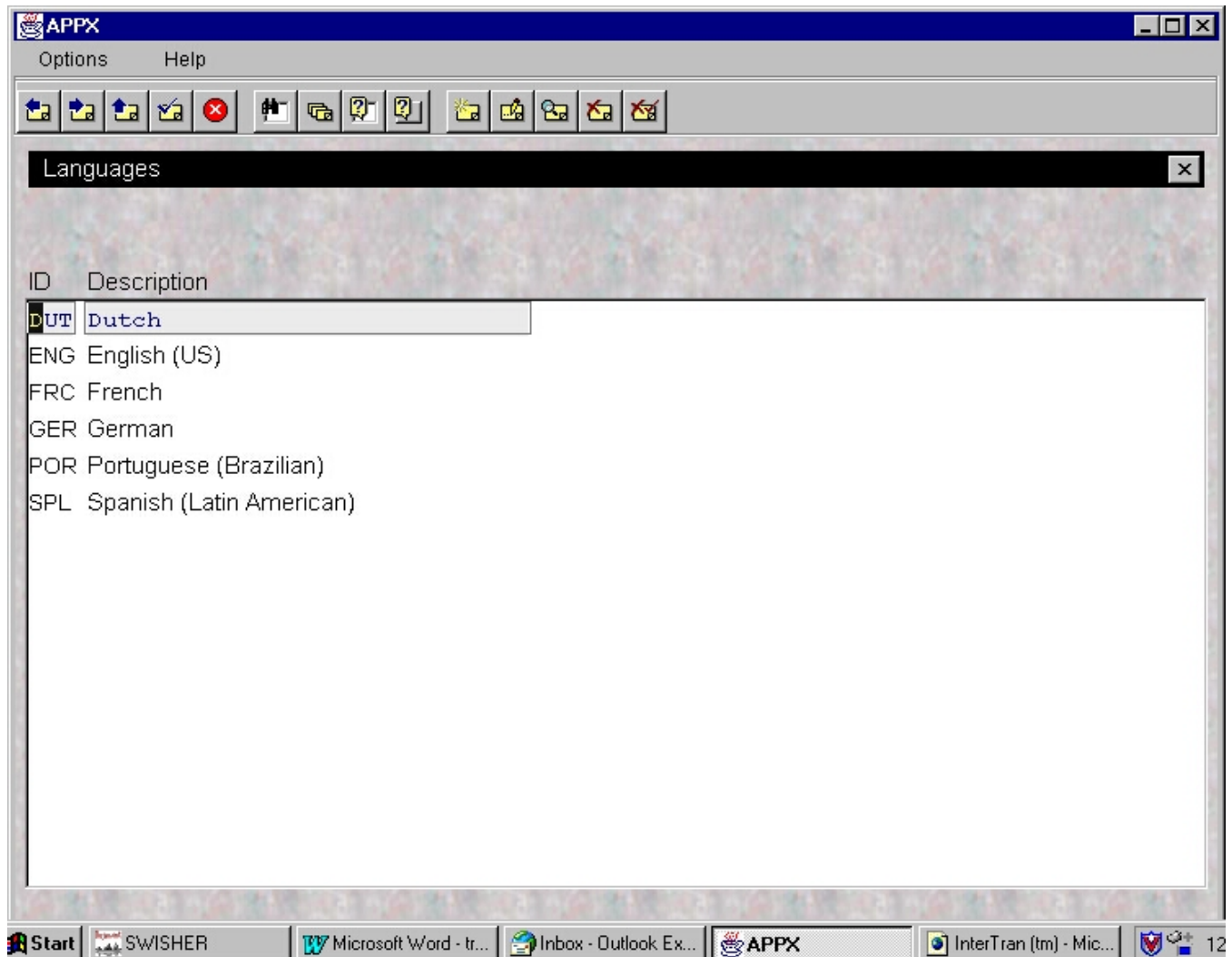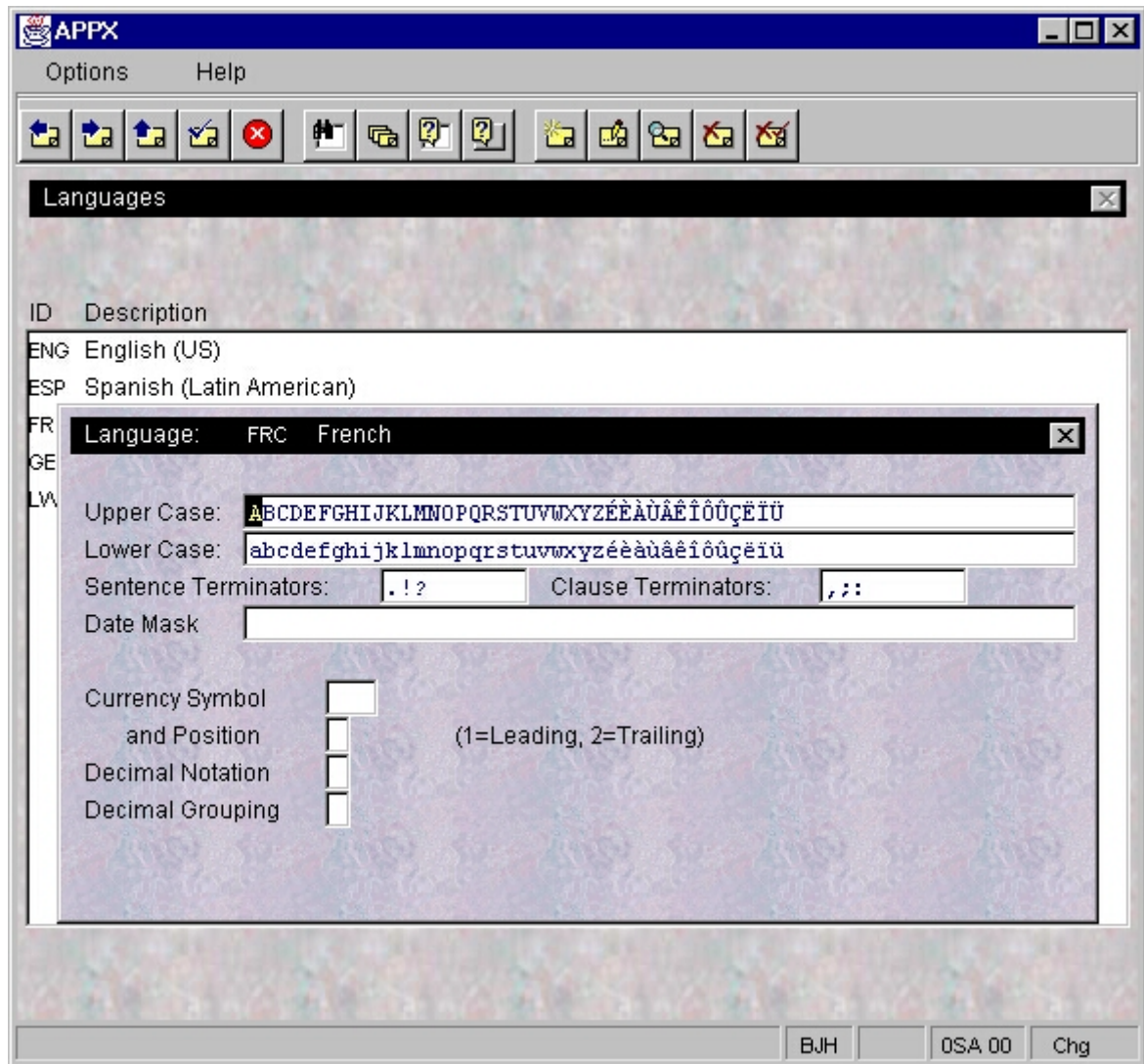

Figure 2 – The Languages Input Process

Figure 3 – Language Field Detail (French)

appx
THE SYSTEM TRANSLATOR
NAAR DE SYSTEMATIEK VERTALER
EL TRADUCTOR DEL SISTEMA

The System Administration Language Translator is used to define translations at a system level. While these translations may be individually overridden at the application level, they are generic enough to be used system-wide. Most of the translations defined here are normally errors, warnings, and messages that are generated by the APPX engine itself, not the APPX applications.

Making system level translations is easy providing, of course, you are bilingual or have access to someone who can make the translations for you. The translation process consists of two steps:

1) Create the system dictionary entries from the APPX engine (in the Design Language)
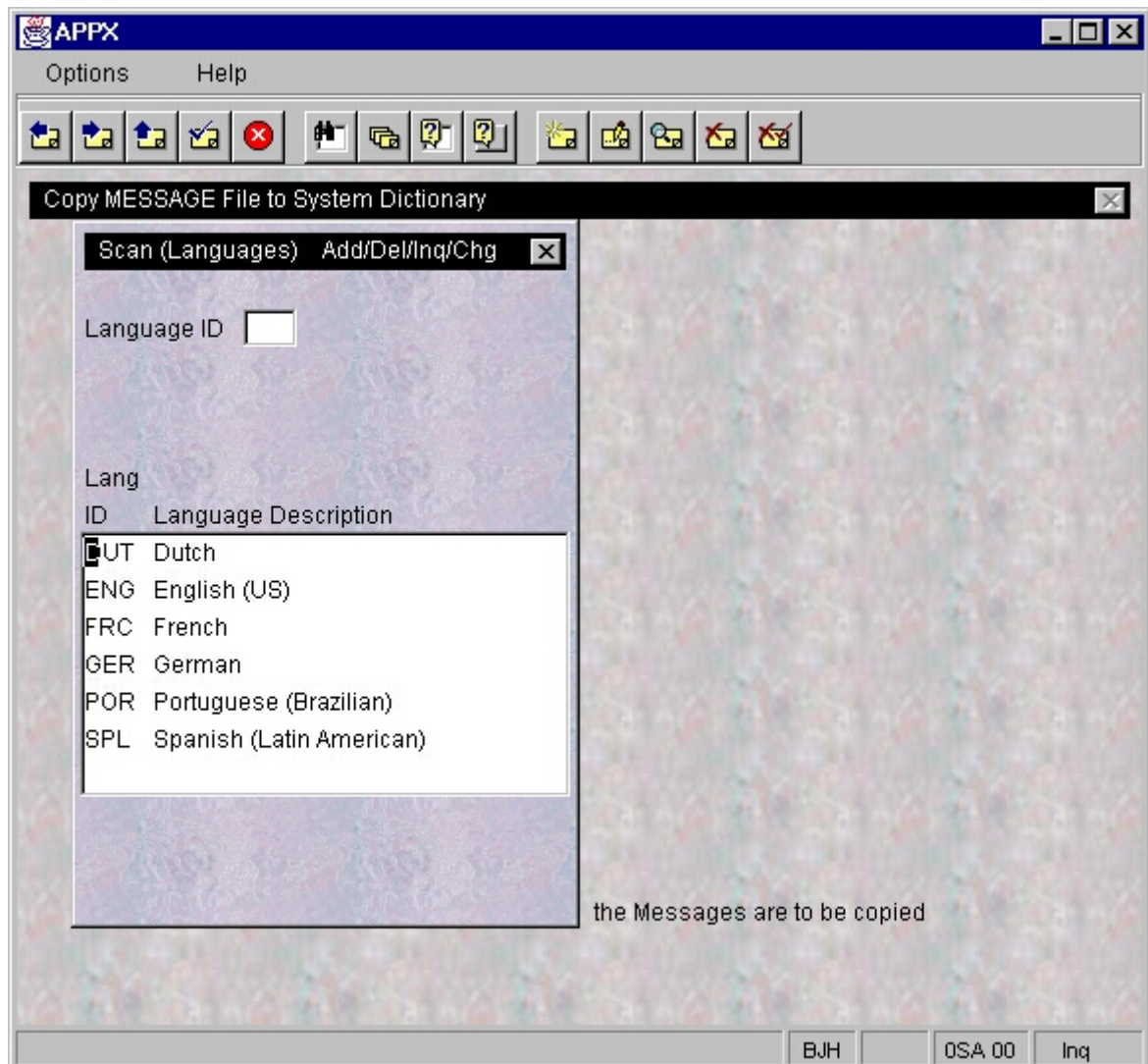2) Define the translations for the dictionary entries created above (in the Presentation Languages)

Figure 4 – Selecting the Translation Language

1. CREATING THE SYSTEM DICTIONARY

Before defining the translations for the APPX System Application, the system dictionary has to be populated with entries for the Design Language (English in this example). Then, corresponding records for the new language (Dutch in this example) are created via the translation process. Until you define these translations, screens continue to be displayed to users in the Design Language, regardless of the Presentation Language they have selected.

From the System Administration's Language Translation Menu, follow these instructions:

➢ **Select** `Copy Message File To System Dictionary.`

➢ To select the Design (or "from") Language , enter the Language ID or *click* the ▼ button to the right of the `Language ID` field box and ***double click*** the code corresponding to this language (English in this example). See *Figure 4.*

➢ **Press** `ENTER` twice (the first time is to display the descriptions of the Language ID's entered; the second time is to confirm the copy request). Wait (normally just a couple of minutes) while the APPX Message File is copied into the Translator Dictionary. Normally, approximately 900 entries (records) are written.
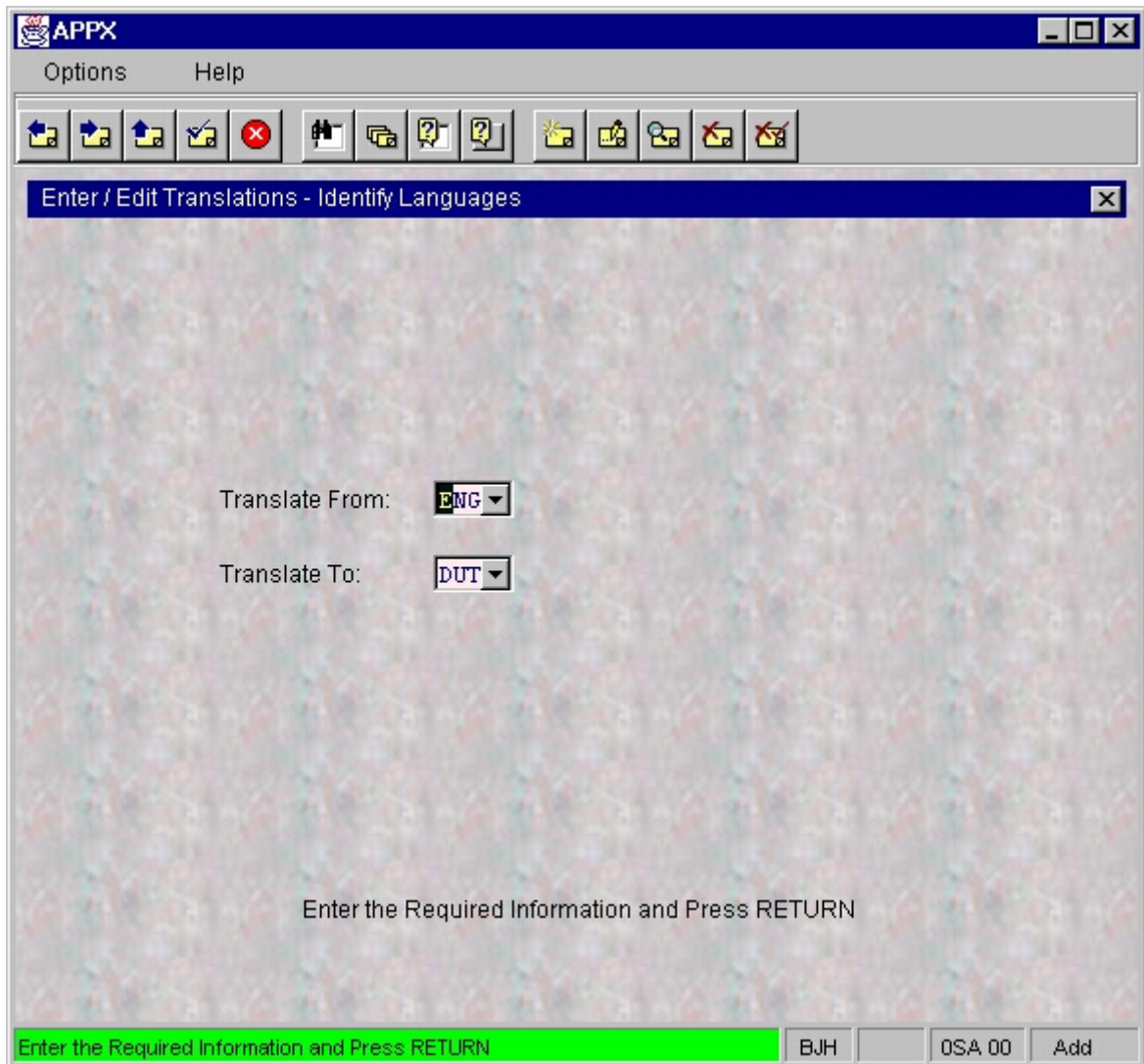

Figure 5 – Identifying Languages For Enter/Edit Translations

2. ENTERING/EDITING TRANSLATIONS

Once the System Dictionary has been populated, you can begin defining translations for your first translated language. These translations are kept in the Translations File and are simply links between two dictionary entries of two selected languages, the "from" language and the "to" language. To create these translations, follow these steps:

➢ *Select* `Enter/Edit Translations` in the Utility section of the System Administration Translation Menu.

➢ When prompted to enter the "From" and "To" languages, **enter** the corresponding language ID's (ENG and DUT, for example. If you do not know the ID's, you can select them from a drop-down list by **clicking** the appropriate ▼ button then **double-clicking** your selection. See *Figure 5*.

➢ *Press* `ENTER` twice to initiate the *Enter/Edit Translations* Input Process.

Figure 6 – Selecting the Dictionary Entry To Translate

➢ The *Enter/Edit Translations* Input Process displays records from the Dictionary File created earlier. See the example of this input in *Figure 6*. Only the first 65 characters of each entry are shown. Use the option buttons at the bottom of the screen to view longer entries. For each "From" record in this file, you must define the translation for the "To" language. It is best to start with the first entry and work your way through the Dictionary File until all entries have been translated. If you want to start with a specific entry, **select** `INQUIRE MODE` first (the other modes are inactive in this process), **type** the first part of the word or phrase you want to translate then **press** `ENTER`.

➢ To translate an entry, **double click** the entry to translate (or **move** the cursor to it and **press** `ENTER` twice) then **click**

> Add Translation

Figure 7 – Entering The Translation

➢ **Type** the translation and **press** `ENTER` when finished.

➢ Continue in this manner until all entries for the "from" language have been translated. As you complete the translation for each entry, a record is written to the Translation File that contains the primary keys of both the "from" and the "to" entries, linking the original text to the translated text. Note that the Dictionary entry in the *Design Language* is now checked as *Translatable*.

To translate to another language, *exit* the program and start it again specifying another "to" language.

AVAILABLE OPTIONS IN ENTER/EDIT TRANSLATIONS

The following options are made available at various points in the *Enter/Edit Translations* Input Process. Just **click** on the selected option.

`View Full Text`	View the entire dictionary text for entries exceeding 65 characters.
`View Full "From" Text`	View the entire translator text for "from" entries exceeding 80 characters.
`View Full Translation`	View the entire translator text for translations exceeding 80 characters.
`Edit Full Translation`	Edit the entire text for translations exceeding 80 characters.
`Select Translation`	Sets the currently highlighted entry as the "to" translation for the current "from" entry by linking the two entries in the Translations File. To use this button, there must be an entry for both the "from" language and the "to" language already present in the Dictionary File.
`Add New Translation`	Allows entry of a new dictionary entry as the translation for the current "from" entry and links the two entries in the Translations File.
`Clear Translation`	"Breaks" the link between the currently highlighted "from" entry and its translation in the selected "to" language.
`Get Def. Translation`	Creates a default translation for the currently highlighted "from" entry. This translation is a literal one based on existing translations for words, clauses, or other smaller components embedded in the current entry. The semi-translated entry is displayed for final edit. Due to the mechanics of this feature, it is recommended that all words and simpler expressions are translated prior to its use.

appx *THE APPLICATION TRANSLATOR*
EL APLICACIÓN TRADUCTOR
EL TRADUCTOR DE APLICACIÓN

With the system level translations completed, you are ready to start translating applications. Each application is translated on an individual basis. You can translate all, several, or even just one application depending on which applications are used in multilingual environments. Perhaps just your order entry application is used in your international offices. If so, it may be the only one you wish to translate. All it takes is four easy steps:

1. Create the translator's design files.
2. Set the application's design language.
3. Create the dictionary from the application's design files.
4. Define translations for the dictionary entries created above.

Ready?

1. CREATING THE TRANSLATOR'S DESIGN FILES

A new set of APPX *Structure Files* must be created for the application if the application was created prior to APPX Version 4.0. These Structure Files identify to the Translator what fields should be translated. Since applications designed prior to 4.0 do not have these fields in them, no translation will occur unless the files are recreated with the new fields in them. You may skip this first step if you are sure the application was created in Version 4.0 or later. If you are not sure what version of APPX the application was created under, perform this step to prevent potential translation problems later on.

➢ From a DOS Prompt:

```
cd c:\appx\data\VV\AAA\Struct        VV=Version        AAA=Application
del *                                Deletes all Structure Files
dir                                  Lists the design files to delete (there should be 23 or 26)
Are you sure (Y/N)?  Y               Confirms the deletion
```

➢ From the APPX Main Menu, *select* `System Administration`, `Database/Applications`, then `Design File Management`.

➢ In the pop-up, *type* the application and version ID of the application you'll be translating. *Press* ENTER.

➢ *Select* `Create Files`. Although most of the application's design files are already created and populated, three files required for the translation feature do not exist if this application was initially created prior to APPX 4.0. Creating all files again will not affect the current design files--each existing file generates a message stating "File Already Exists" while the three new files (DICTNARY, XLAT, EXCEPT) are created.

2. SETTING THE DESIGN LANGAUGE

Only one version of an *Application Design* is maintained. These program, or design, files are written in what has been designated the Design Language. A Design Language must be defined for each application if translation is to occur. To set the Design Language:

➢ *Select* `Application Design`, `Utilities`, `Application Parameters`.

➤ **Enter** the `Base Language ID` and **press** `ENTER` to return.

> WARNING: If you do not perform this step then the following step, creating the dictionary from the applications design files, will appear to run properly but will not, however, add any entries to the dictionary because the process is unable to determine what this application's design language is.
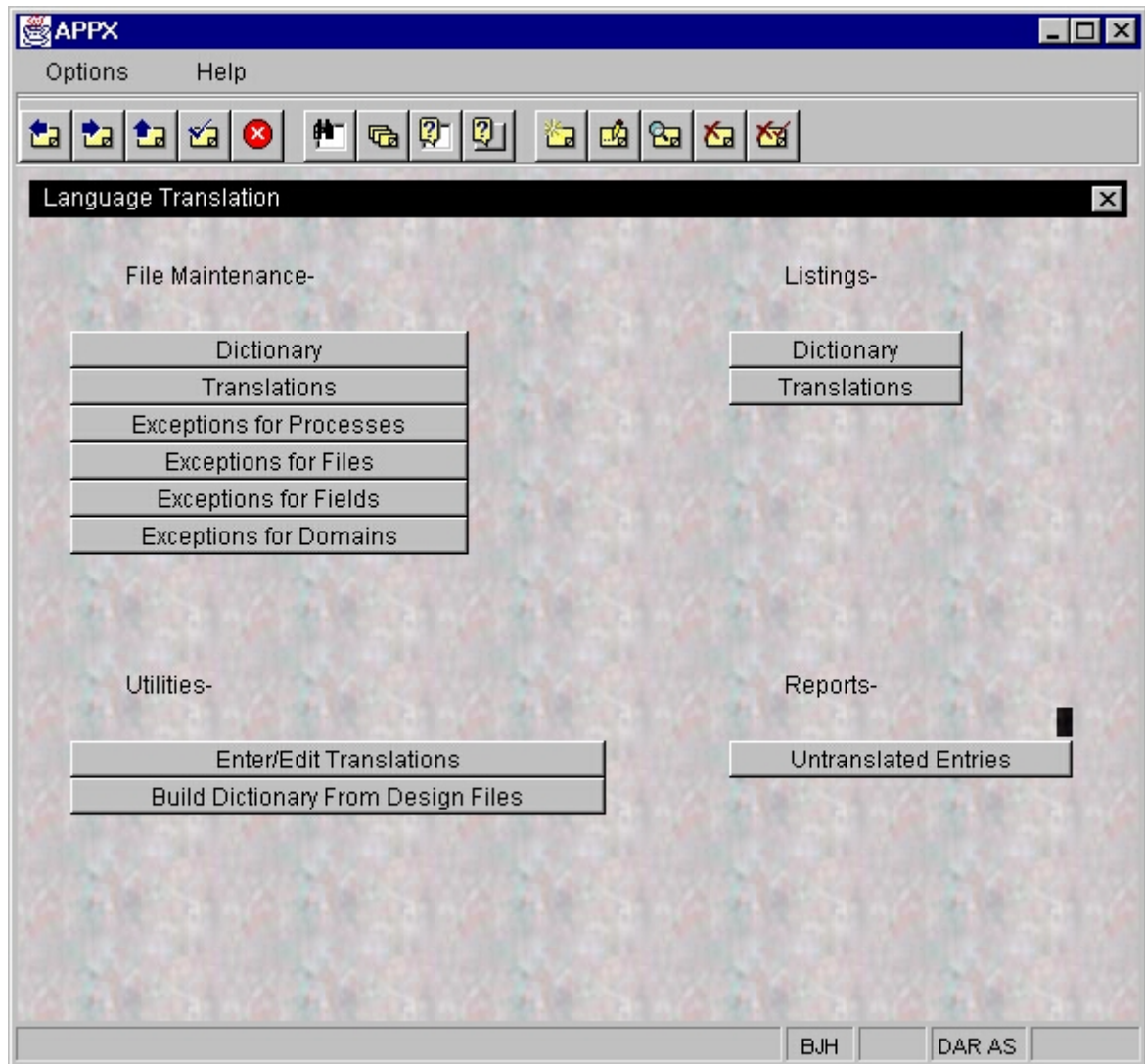

Figure 8 – The Application's Language Translation Menu

3. CREATING THE DICTIONARY FROM THE APPLICATION'S DESIGN FILES

The next step is to create the dictionary entries from the existing design files. This process parses the design files, creating entries in the application's dictionary for each text, paragraph, sentence, clause, and word. Just follow these steps to create the dictionary:

➤ From the `Application Design Menu` for the application you are translating, **select** `Utilities, Language Translation,` then `Build Dictionary From Design Files` (see *Figure 8).*

➢ To select the Design (or "from") Language , enter the Language ID or *click* the ▼ button to the right of the `Language ID` field box and *double click* the code corresponding to this language.

➢ *Press* `ENTER` twice (the first time is to display the descriptions of the Language ID's entered; the second time is to confirm the copy request). Wait (normally just a couple of minutes) while the Dictionary File is created from the application's design files.

When the dictionary entries are created, you can get a count of the number of entries in the file by returning to the APPX Main Menu. *Select* `Database/Applications`, the application you are translating, `Design File Management`, `Verify Files Menu`, then `Verify Integrity of APPXIO Files`. The entry count is the record count displayed for the DICTNARY file and indicates the number of entries requiring translation.

4. ENTERING/EDITING TRANSLATIONS

Defining translations at the application level is identical the steps used at the system level. From the application's Language Translation Menu, *select* `Enter/Edit Translations` and follow the steps outlined under *Entering/Editing Translations* in Section 3.

AVAILABLE OPTIONS IN ENTER/EDIT TRANSLATIONS

All options available at the system level are also available here.

appx SOFTWARE THE DICTIONARY FILE
DIE LEXIKON AKTE
EL ARCHIVO DEL DICCIONARIO

THE DICTIONARY FILE INPUT PROCESS

A *Dictionary File* is maintained at the System Administration level. Dictionary Files are also found in all applications using the Language Translator feature with each application having its own file. The Dictionary Files are populated with Design Language entries by Language Translator update processes at both the System Administration and the Application level. Entries in other languages (the Presentation Languages) are added to the files by use of the Enter/Edit Translations input process, or by the Dictionary File Input Process. The Dictionary File Input Process is shown in *Figure 9* and an explanation of the Dictionary File fields is shown in *Figure 10.*

Figure 9 – The Dictionary File Input Process

FIELD	DESCRIPTION
Language ID	Corresponds to Language Codes and their descriptions in the System Administration Language File.
Hash ID	A hash number calculated from numeric values assigned to each character in the dictionary entry text. This, along with the Language ID and Sequence Number, creates a unique key for the entry.
Sequence No	Usually zero unless more than one dictionary entry produces the same Hash ID. This is the case with identical entries using different capitalization. For example, the entries "ERROR" and "error" will produce the same HASH ID but the second entry will be assigned a Sequence Number of 01 to give the entry a unique primary key.
Text Type	Indicates the level of dictionary entry as the translation process looks first for matches in the largest to smallest portions of the text being translated. See *Figure 12* for a list of Text Types.
Translatable?	Checkbox indicating whether the entry can be translated or not. For example, "APPX" has no equivalent in most languages and therefore, no attempt should be made to translate it. This box is automatically checked when a translation is made for an item.
Case Sensitive?	Checkbox indicating if an exact case match must be made between this entry and the text to be translated in order for this entry's translation to be used. For example, if the translator is attempting to translate "An Error Has Occurred" and finds the dictionary entry "AN ERROR HAS OCCURRED" it will use the entry's translation if it is not Case Sensitive. If it is, the translator will attempt to find an exact match and if it fails, it will not provide a translation for the text. In most cases, entries should not be marked as Case Sensitive unless an extract entry with different capitalization exists.
Abbreviation?	Checkbox indicating if the entry is an abbreviation or not.
Length	The number of characters contained in the entry (embedded spaces included).
Text	The dictionary entry.

Figure 10 – The Dictionary File Fields

AVAILABLE OPTIONS IN THE DICTIONARY FILE INPUT PROCESS

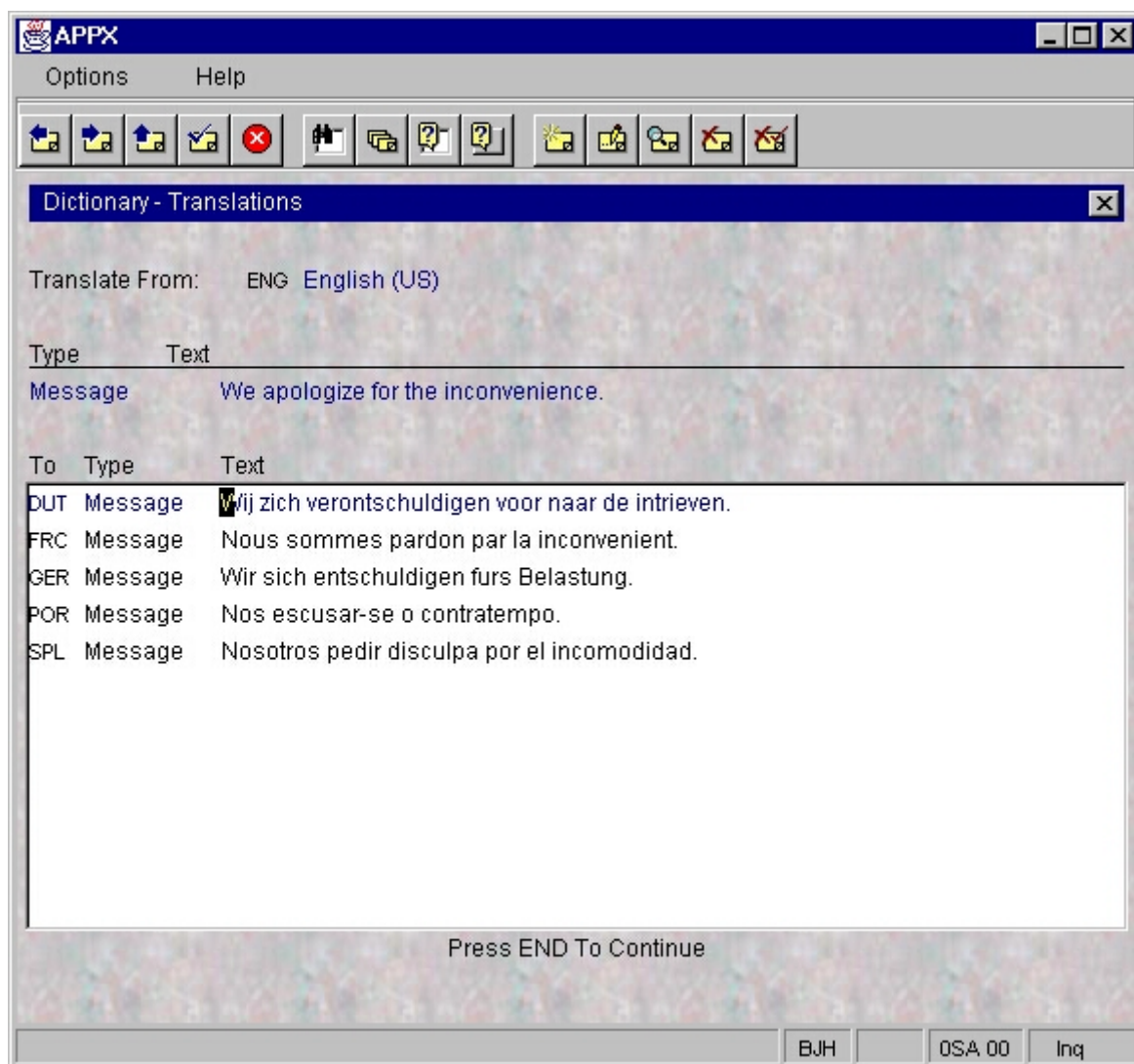|	
Edit Full Text	Edit the full entry for entries exceeding the 40 characters displayed in the scrolling portion of the frame.
Show Translations	Displays the corresponding dictionary entries in other languages that are linked to the currently displayed entry. See the example in *Figure 11*.

Figure 11 – The Translations Linked To A Dictionary Entry

TYPE	DESCRIPTION
Word	A single word or two words joined by a hyphen.
Phrase	A combination of words not complete enough to be classified as a clause or sentence.
Clause	A group of words ending with one of the Clause Terminators defined for this Language in the Language File.
Sentence	A group of words ending with a Sentence Terminators defined for this Language in the Language File.
Paragraph	A group of sentences.
Text	A group of paragraphs.
Quote	A group of words preceded by and terminating with a double quotation mark.
Field	The *Token* or *Validation Table* value of a File or Work field or the value of a Work Field defined as Alpha or Text. In the Data Dictionary, make sure the *Translatable?* checkbox for this item is checked for translation to occur. It is recommended that a Work Field defined as Translatable be non-modifiable on Input Processes.
Row	A complete or partial row of an APPX image.
Message	An error, warning, or message generated by the APPX *ILF Editor*.
Token	Allowable values for fields defined as token fields.
Exception	Any of the preceding text types that has an exception, or override translation.

Figure 12 – Dictionary Entry Types

THE TRANSLATIONS FILE
NAAR DE VERTAALWERK VIJL
EL ARCHIVO DE TRADUCCIONES

The *Translations File* is used by the translator as a kind of cross-reference. Its primary index is the Design Language ID, the dictionary entry's *Hash ID*, and the dictionary entry's *Sequence Number*. After parsing screens into paragraphs, sentences, clauses, and words, the Translator finds these components in the Dictionary File using the Design Language ID and the component's Hash ID. Then, it looks to the Translations File to find the record for the current entry in conjunction with the Presentation Language ID.   After locating this record, the translator has the primary index of the dictionary's corresponding entry in the Presentation Language. The Dictionary File is read again using that index and the text for the Presentation Language is captured by the translator.

Figure 13 – The Translations File Input Process

The *Translations File* can be edited by ***selecting*** `Translations` under `File Maintenance` on either the *System Administration* or *Application Language Translator Menu*.

NOTE: The addition, modification, or deletion of translations is best performed via the Enter/Edit Translations Input Process found on the Language Translator Menu on either the System Administration or Application level. The Enter/Edit Translations process provides the most comprehensive and concise method of performing translation maintenance.

As shown in *Figure 13*, the *Translations* Input Process is very basic. The "from" and "to" entries are shown along with the respective primary keys in the Dictionary File. You can change the translation "to" or "from" entry by selecting a different Hash ID/Sequence Number combination. Use the ▼ button to the right of the appropriate Hash ID field to scan the Dictionary File for other available entries.

The Translator locates the appropriate translation by searching for the text in the application's Exception File. If no match is found, the Translator then searches for various components of the text in this order:

1) Full screen text
2) Paragraphs
3) Sentences
4) Clauses
5) Words

At each of these five levels, the translator looks for matches in these files in the order shown:

1) The current application's *Exception File*
2) The current application's *Dictionary* and *Translations Files*
3) The system level *Dictionary* and *Translations Files*

If no match is found, the text is presented in its original format with no translation occurring.

appx SOFTWARE

THE EXCEPTIONS FILE
ECCEZIONE SCHEDARIO
EL ARCHIVO DE EXCEPCIONES

The Exception File consists of translation exceptions defined by the application designer. Most translations are defined via the Dictionary and Translations Files as described in previous sections. There are times, however, when the designer may wish to override the translation for a specific word, clause, or sentence. For example, a sentence might not have a translation defined for it but the Language Translator is able to translate its component words. While the translation is successful, it might result in grammatical nonsense that requires a more meaningful translation to provide a polished presentation. Exceptions might also be defined for sentences or text that the Language Translator can only translate in fragments, resulting in text that displays with only a partial translation.

Exceptions can be defined and maintained using these methods:

1)   Via *Exception File Input Processes* found on the application's *Language Translation Menu*

The preferred method when you have a number of exceptions to define and plan on dedicating a block of time to define them. This method provides for a means of sequentially accessing only the items for which exceptions can be made.

2)   Directly from *Application Design* while in *Data Dictionary Management* or *Process Management*

This is the most efficient method when designing new applications that were not originally part of the application's initial Dictionary File build and translation process. It is also recommended when defining exceptions for a limited number of items. For the bi- or multi-lingual designer, it is more efficient to define translations as exceptions on the fly while designing than to later add items into the Dictionary File and define translations in the Translations File.

Before learning *how* to make exceptions, it is first important to learn *when* (i.e. what design components) exceptions can be made. Take a look at the following chart:

Design Component	Location
In Progress Message	Job, Query, Output, Update, Status
Report Title	Output, Inquiry
Field Description	Domain, Field, Work Field
Column Headings	Domain, Field, Work Field
Tokens, Validation Table Entries	Domain, Field, Work Field
User Documentation	All Processes, Domain, Field, Work Field
Image Text (single row only)	Input, Output, Inquiry

Figure 14 – Exception Components

DEFINING EXCEPTIONS VIA THE EXCEPTION FILE INPUT PROCESSES

Four Input Processes can be found on the Application Language Translation Menu (see *Figure 8).* Each of them provide you with a means of going through all of the application's processes, files, fields, or domains in sequential order to define or maintain exceptions. Using *Figure 15,* determine what components exceptions can be defined for in each of the four Exception File Input Processes.

Exception File Inputs	Components Exceptions Can Be Made For
Processes	• In Progress Message (Job, Query, Output, Update, Status) • Report Title (Output, Inquiry) • Documentation (User only) • Single Row of Image Text (Input, Output, Inquiry)
Files	• User Documentation
Fields	• Field Description • Column Headings • Tokens, Validation Table Entries • User Documentation (User only)
Domains	• Same As Fields

Figure 15 – Exception File Input Processes and Exception Components

To define exceptions using the Exceptions File Input Processes, follow these steps:

➢ On the application's *Language Translator Menu*, **select** the process under *File Maintenance* that corresponds to the exceptions you want to make.
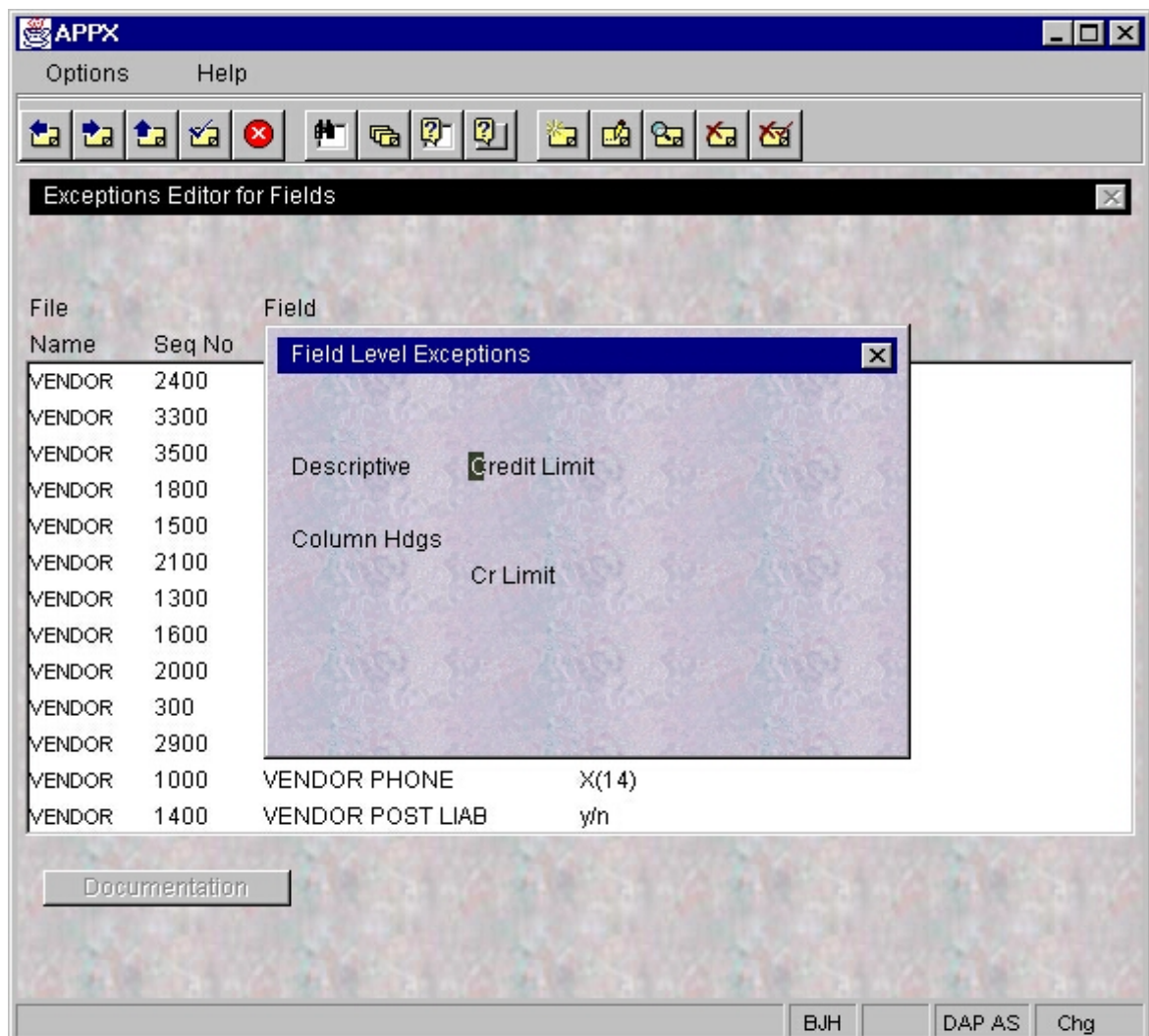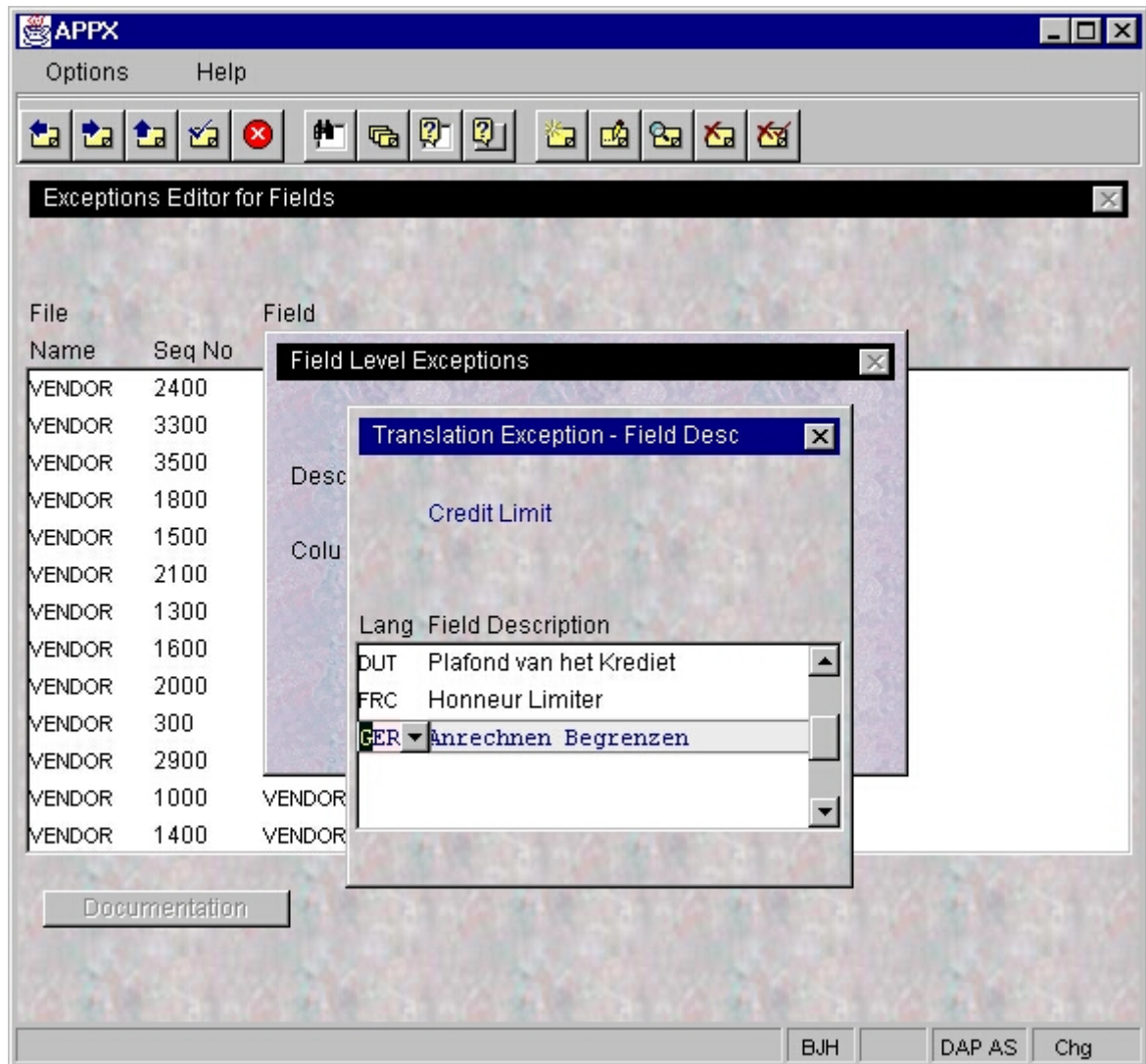
Figure 16 – The Field Level Exceptions Window

Figure 17 – Translation Exceptions For A Field Description

➢ The process displays a list of design processes, domains, fields, or domains depending on the input process you are running. From this scrolling display, each process has different options in the form of buttons that are available at the bottom of the screen.

Documentation	Displays the User Documentation in the Design Language for this process, file, field, or domain. ***Click*** this button if you want to define User Documentation exceptions.
Process Fields	Displays the In Progress Message and Report Title in the Design Language for processes. ***Click*** this button if you want to define exceptions for either process field.

OR

Press ENTER to drill down to an image (for Processes) or to display the Field Description and Column Headings (for Fields and Domains).   You must display the fields prior to defining exceptions for them. See *Figure 16*.

➢ When you are positioned at the field--or within a row of image text-- you want to define an exception for, *select* OPTION 95 (OPTION, OPTION, 9, 5). This option initiates the Exceptions File Editor.

➢ If you are defining an exception for a row or partial row of image text, the editor instructs you to move the cursor to the last character in the string of text (or row) you are defining an exception for. See *Figure 18*. **Move** the cursor to the end of the string or row and *press* ENTER. If you are not defining an exception for a row or partial row, skip this step.

➢ The editor needs to know the Presentation Language you are defining the exception for. **Enter** the Language ID (or *click* ▼ to *select* from a drop-down list) then *press* ENTER twice.

➢ *Select* the appropriate *Mode (ADD MODE, CHANGE MODE or DELETE MODE) d*epending on whether you want to add a new exception, change an existing one, or delete an unnecessary one.

➢ *Type* the exception's translation in the presentation language you indicated. The exception can also be modified or deleted at this point.

➢ When finished, *press* ENTER to save the translation (or ACKNOWLEDGE DELETE to delete it).

➢ *Enter* translations in additional Presentation Languages, *pressing* ENTER after each. See *Figure 17*. Or , you can modify/delete existing exceptions.

➢ When all translations are defined, modified or deleted for this field or image, *select* END until you return to the scrolling display of processes, files, fields, or domains.

DEFINING EXCEPTIONS VIA APPLICATION DESIGN

Exceptions can also be defined and maintained directly from *Application Design*. This maintenance is performed in either *Data Dictionary Management* (for *Domains, Files*, and *Fields)* or in *Process* Management (all processes).

Using *Figure 14*, determine what items exceptions can be made for. Then define, modify, or delete exceptions by following these instructions:

➢ When you are positioned at the field--or within a row of image text-- you want to define an exception for, *select* OPTION 95 (OPTION, OPTION, 9, 5). This option initiates the Exceptions File Editor.

➢ If you are defining an exception for a row or partial row of image text, the editor instructs you to move the cursor to the last character in the string of text (or row) you are defining an exception for. **Move** the cursor to the end of the string or row and *press* ENTER. See *Figure 18*. If you are not defining an exception for a row or partial row, skip this step.

➢ The editor needs to know the Presentation Language you are defining the exception for. **Enter** the Language ID (or *click* ▼ to *select* from a drop-down list) then *press* ENTER twice.

➢ *Select* the appropriate *Mode (ADD MODE, CHANGE MODE or DELETE MODE)* depending on whether you want to add a new exception, change an existing one, or delete an unnecessary one.

➢ *Type* the exception's translation in the presentation language you indicated. The exception can also be modified or deleted at this point. (See *Figure 19).*

➢ When finished, *press* ENTER to save the translation (or ACKNOWLEDGE DELETE to delete it).

➢ *Enter* translations in additional Presentation Languages, *pressing* ENTER after each. Or , you can modify/delete existing exceptions.

➢ When all translations are defined, modified or deleted for this field or image, *select* END to exit the *Exceptions Definition Process* and return to *Data Dictionary Management* or *Process Management*.
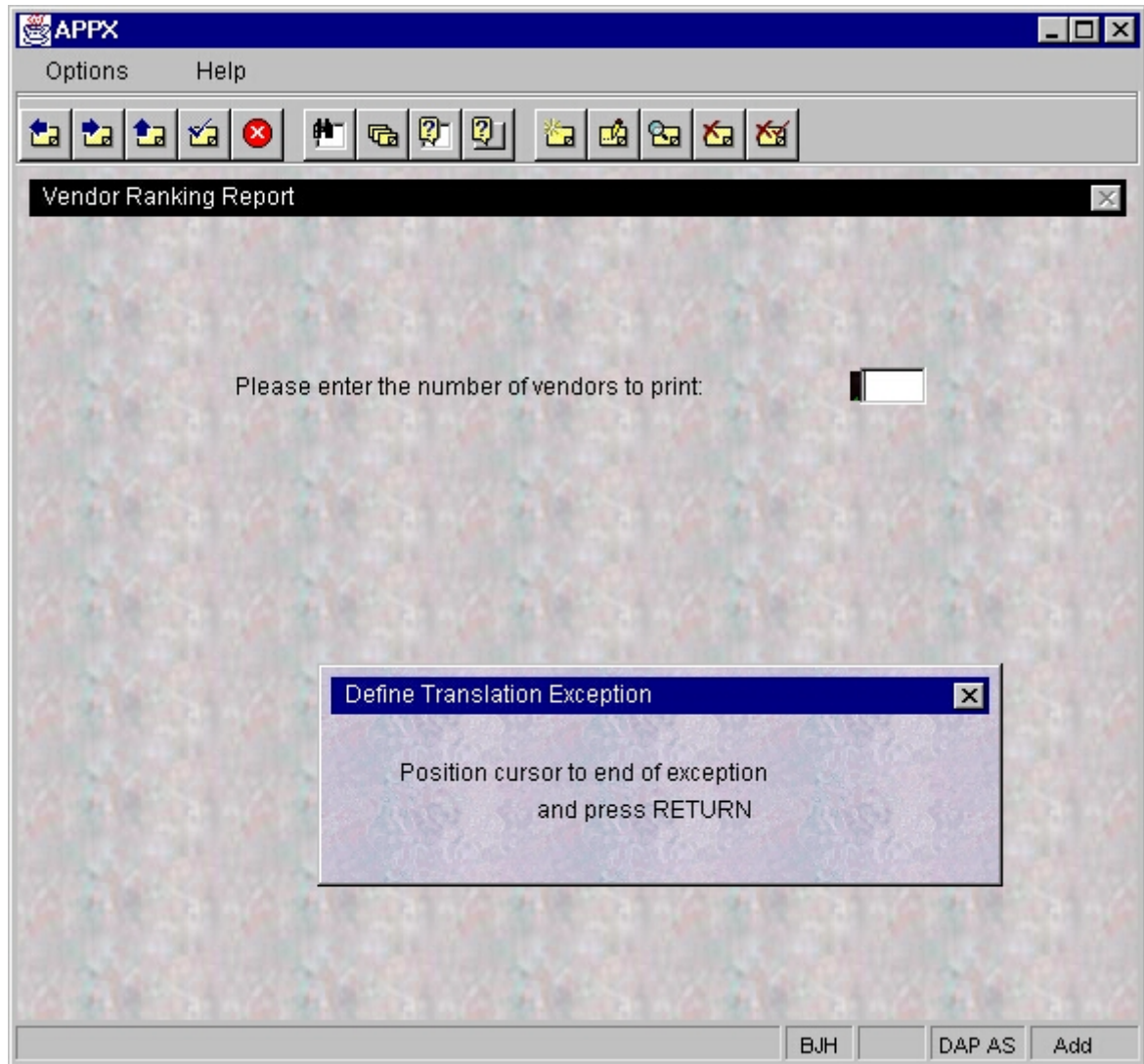

Figure 18 – Selecting an Image Row or Partial Row to Define an Exception For

WARNING: It is important to note that truncation may occur if the translated text is longer than the original design language text. This could result in incomplete translations being presented to the user as data fields are not shifted right on images to accommodate translations that exceed the length of the original text. To avoid this, make sure enough space exists on the image for all of the item's translations to fit.
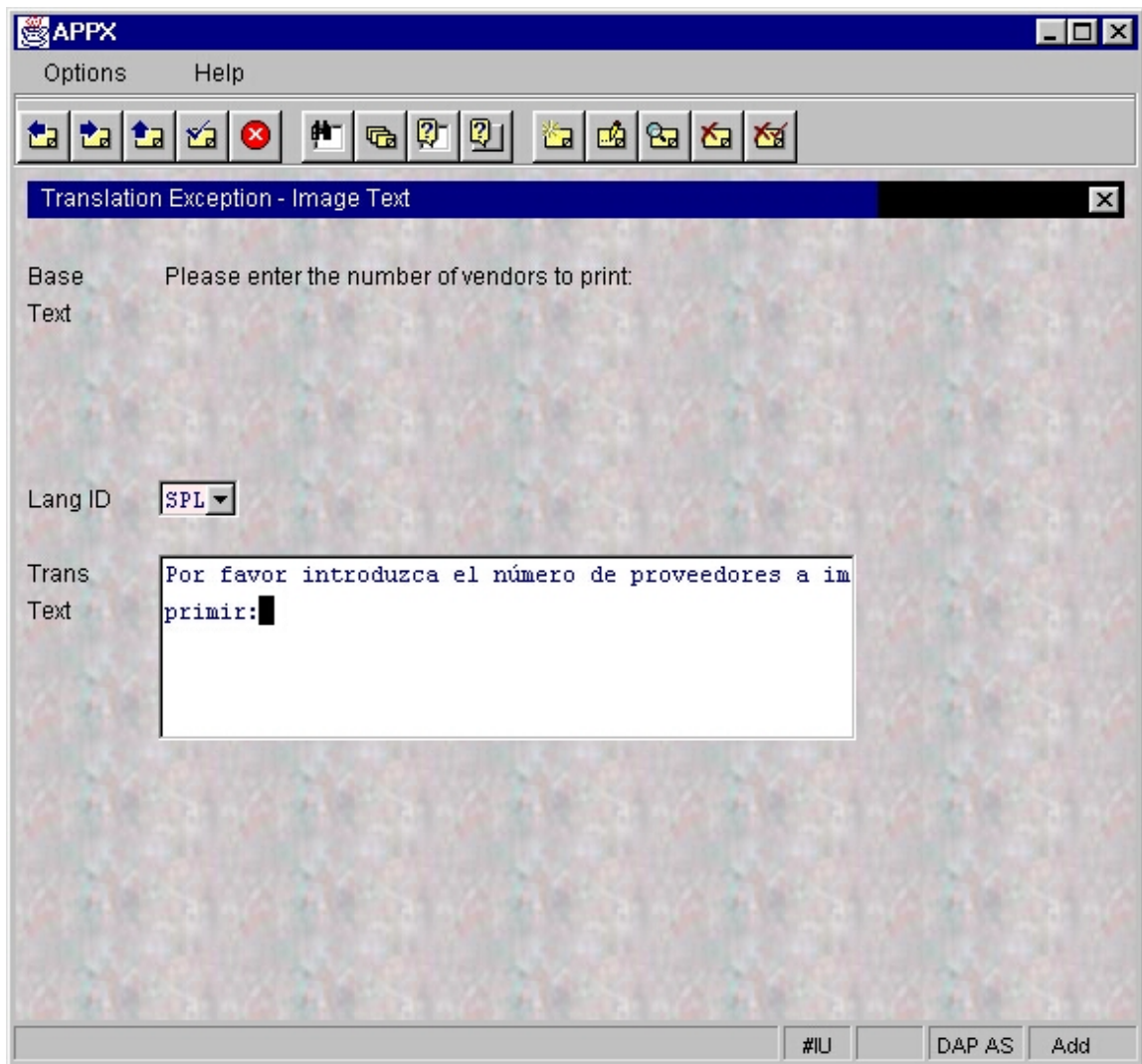
Figure 19 – Defining an Exception For an Image Row or Partial Row

appx *SELECTING THE PRESENTATION LANGUAGE*
UITGEZOCHT NAAR DE VOORSTELLING TAAL
SELECIONANDO A APRESENTAÇÃO LÍNGUA

During runtime, the application is presented to users in one of the Presentation Languages. Much of the translation occurs when the Executable Module is built for a process, although some translation may also occur during actual runtime. The Design Language and default Presentation Language for users is specified at the system level, within System Administration. The Presentation Language, however, can be overridden at several different levels.

These Presentation Language overrides are assigned a specific hierarchy with each successive level superceding the previous level. See *Figure 20* for details.

Level	Language ID Field Location
1. System Related Applications	`System Administration, System Setup, System Parameters`
2. Database	`System Administration, Database/Applications`
3. Database/Application	`System Administration, Database/Applications, Databases/ Applications, System Related Applications`
4. User	`System Administration, Security, Users`
5. Dynamically at runtime	Predefined Field: --- NEXT LANGUAGE ID

Figure 17 – Translation Override Levels

❑ For example, the Triple Z company's two APPX applications are designed in English but most corporate employees speak Latin American Spanish. Therefore, in the System Parameter File, the default Language ID should be defined as `SPL`.

❑ Both applications are written in English. Therefore, in each of the Applications' Parameter File, the default Language ID should be `ENG`.

❑ The Triple Z company is based in Columbia where most users have a native language of Latin American Spanish. Their single database (ZZZ) should be assigned a default Language ID of `SPL` or, can be left blank as the System Language is already defined as Spanish.

❑ Triple Z's accounts payable department, however, is located in Brazil where all its employees speak Portuguese. The accounts payable application (DAP) within the ZZZ database should be assigned a default Language ID of `POR`. See *Figure 21*.

❑ Marie is a new accounts payable clerk from Martinique who speaks fluent Portuguese but has difficulty reading it. Therefore, she has requested that all of her screens be presented in her native French. In the User File, Marie's default Language ID should be defined as `FRC`. See *Figure 22*.

❑ Then there is user OTL. With three distinct personalities each fluent in a different language, this user needs to be able to change their display from one language to another with a second's notice. One minute the user is Oriel, a French mother of two young children. Then minutes later there's Tresa, a German college student.   She takes over and can only read German. And sometimes Luis, a Latin dance instructor by night, is sitting in front of the workstation. He must have everything in Spanish. Of course.

A nightmare for OTL's mental health professional, maybe. But not for you, the APPX designer. By using the predefined field NEXT LANGUAGE ID, menus can be designed with built-in optional processes that automatically change the Presentation Language to a selected language (see *Figures 23 and 24*). Click the `Español` button to run a single statement Subprocess that changes the NEXT LANGUAGE ID to SPL via the SET command. Click the `Duetsch` button to run the Subprocess that changes the language to GER and click `English` to change back to English.

```
SET    ---  NEXT LANGUAGE ID              =    SPL
```
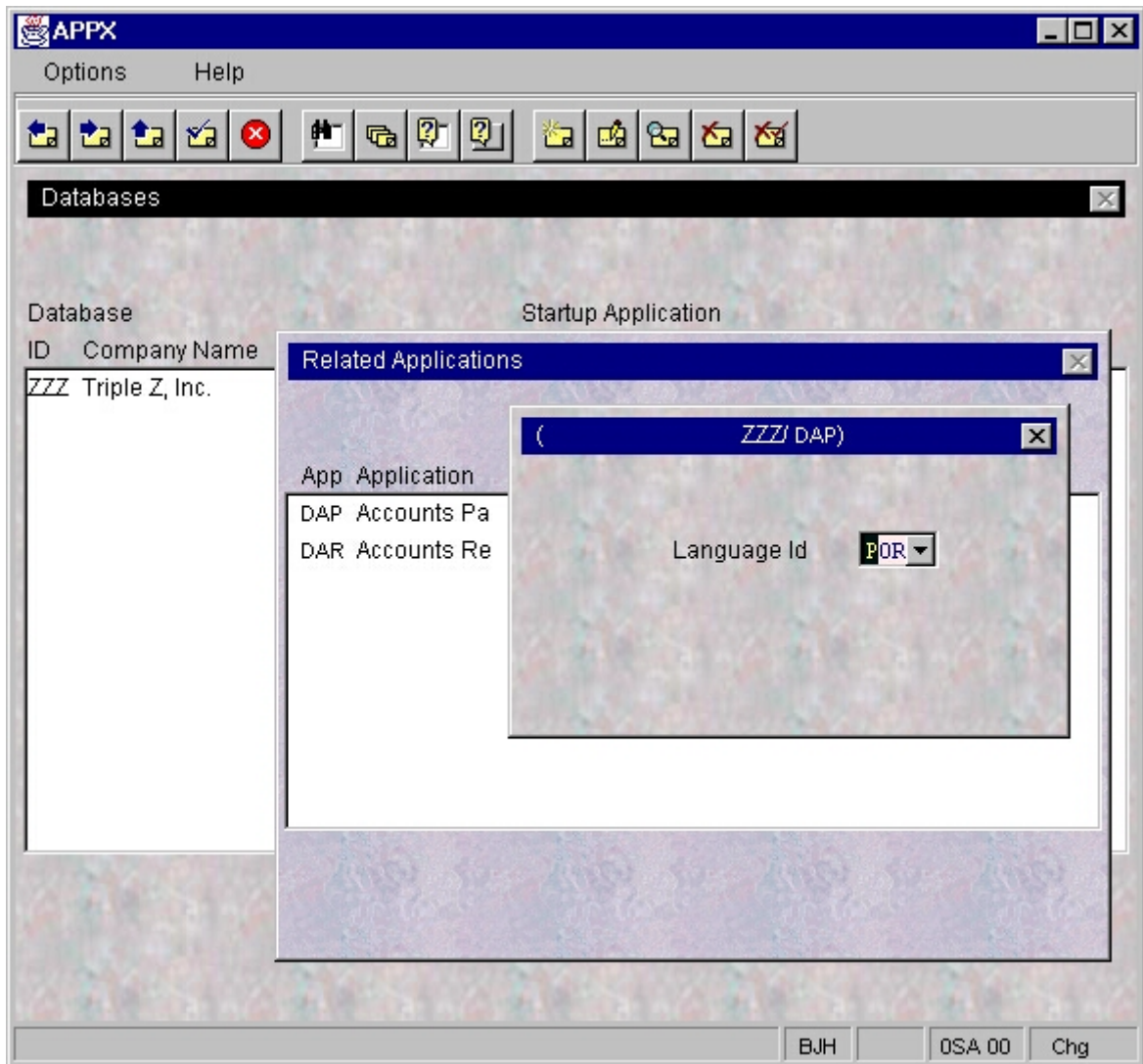

Figure 21– Defining A Language For An Application Within A Database

WARNING: When creating/loading an Executable Module or when performing a runtime translation, a translation will occur only if the designated Design Language (in the application's Parameter File) **and** the designated Presentation Language (one of five settings in the five-level Presentation Language hierarchy) are **both** non-blank **and** not equal to each other. If either is blank **or** if they are both the same Language ID, no translation occurs.
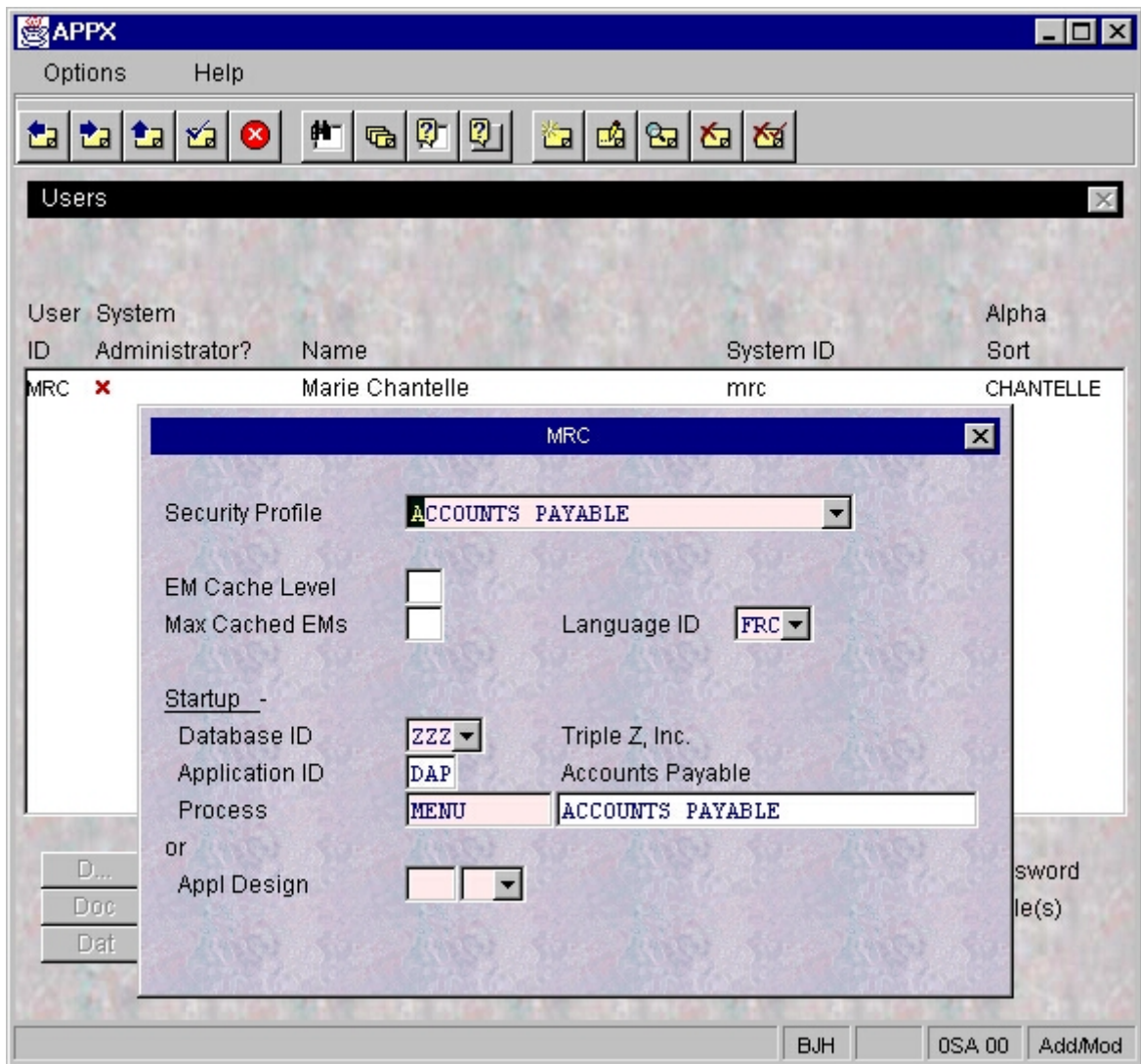
Figure 22 – Defining A User's Presentation Language

Figure 23 – Using Optional Processes to Change The Presentation Language

NOTE: Each of the above Subroutine Processes consists of a single statement

SET   --- NEXT LANGAUGE ID   =   LLL

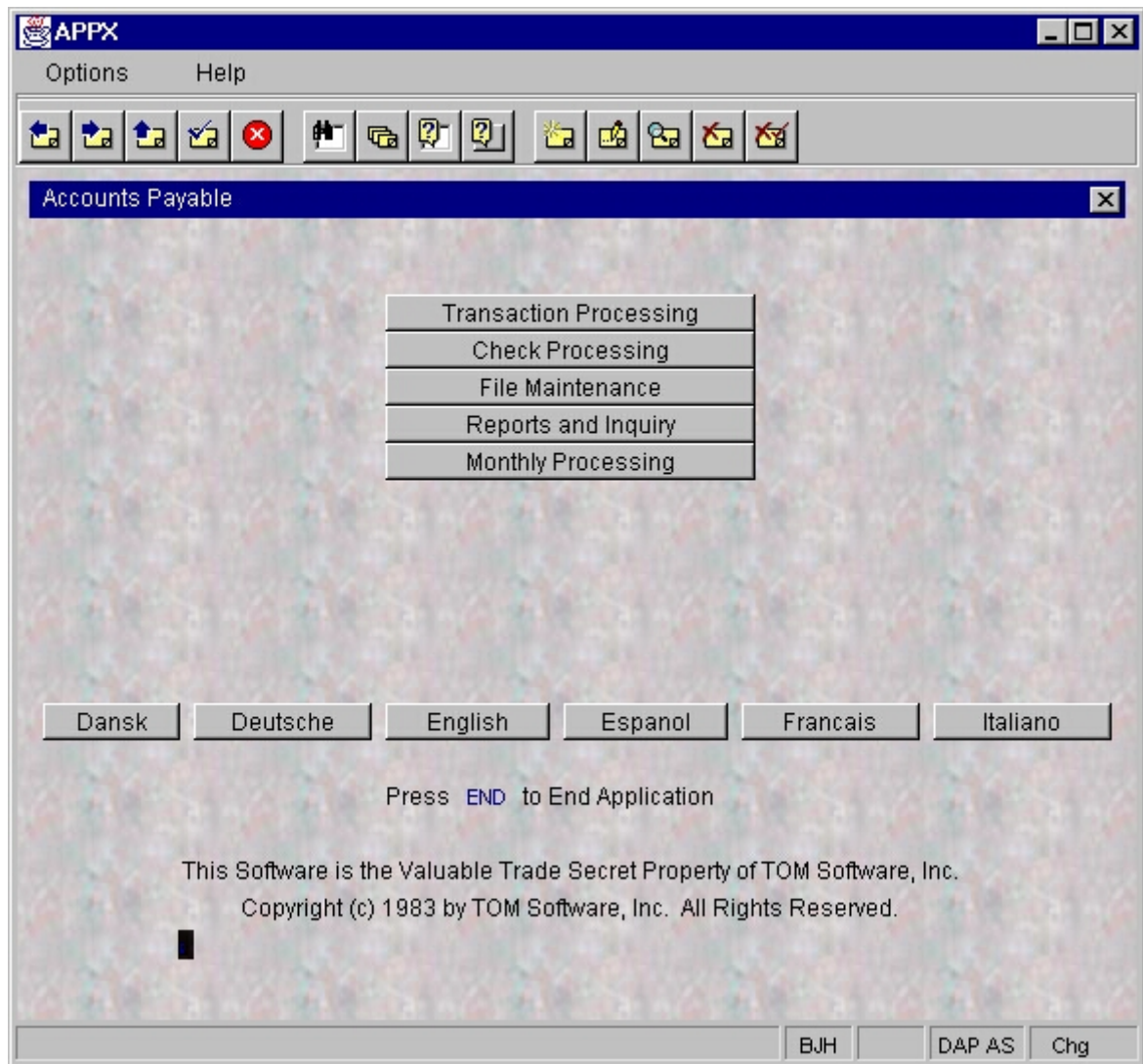Where LLL equals the corresponding Language ID

Figure 24 – Offering Runtime Presentation Language Changes

appx DESIGN TECHNIQUES
ENTWERFEN TECHNIKEN
PLAN TECHNICAL

To assist application designers in the translation process, APPX provides you with tools that can be used from *Application Design*. These tools can be initiated by selecting the appropriate option:

OPTION 17: DISPLAY/HIDE TRANSLATION EXCEPTIONS

This option acts as a toggle switch to mark or unmark all portions of the image that have *Translation Exceptions*.

When viewing an image in the *ILF Editor*, **select** OPTION 17 (OPTION, OPTION, 1, 7) to highlight all portions of the image for which *Translation Exceptions* have been defined.

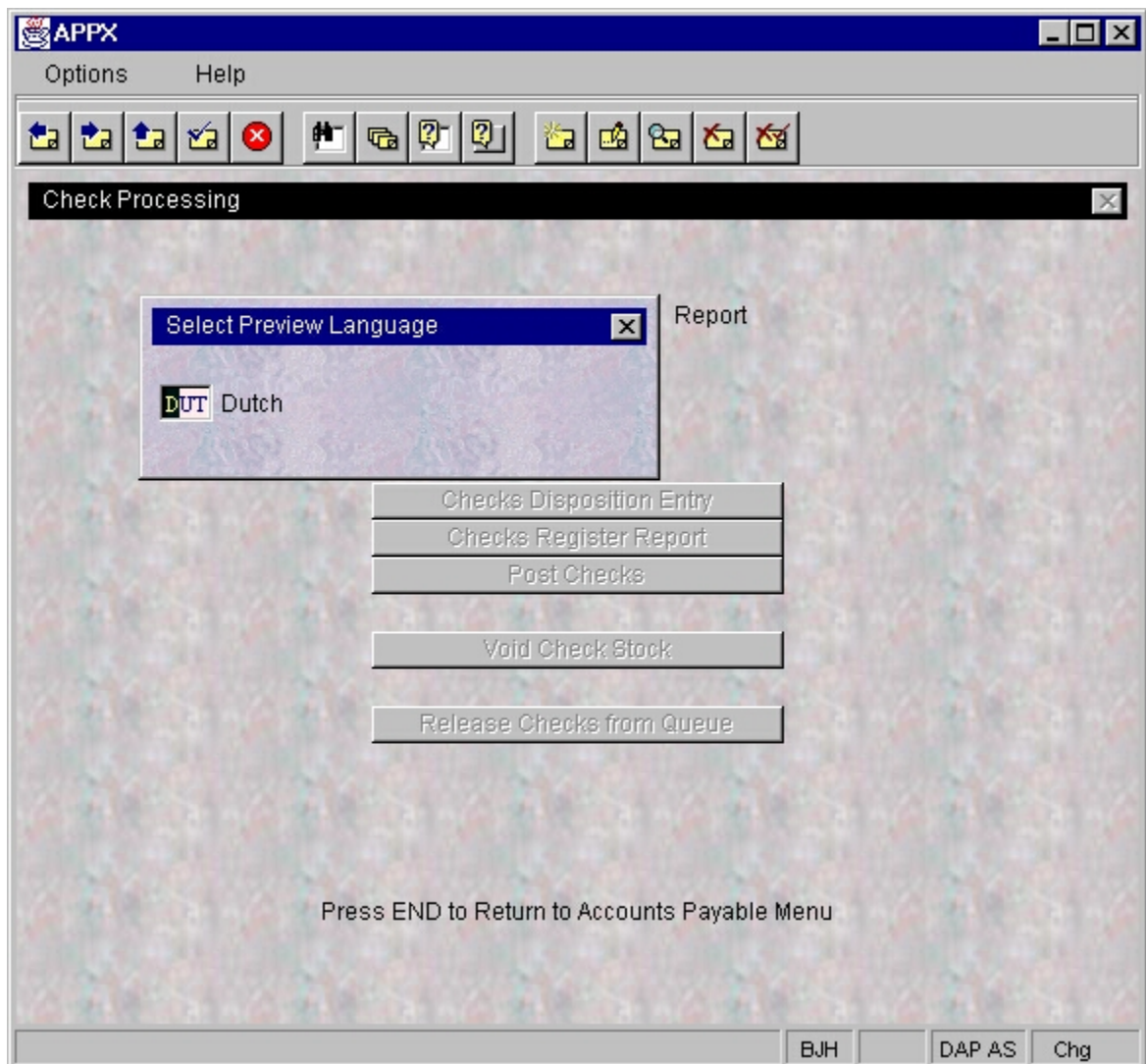To restore the image back to its unmarked state, **select** OPTION 17 again.

Figure 25 – Selecting The Preview Language

OPTION 94: THE PRESENTATION PREVIEW

The Preview feature displays Menus and Input Processes in any of the defined Presentation Languages you select. While in the Image Editor, follow these steps to preview the image in another language:

➢ From anywhere on the image, **select** `OPTION 94 (OPTION, OPTION, 9, 4)`.

➢ *In the Select Preview Language pop-up window (see Figure 25),* **enter** the Language ID of the Presentation Language you want the image translated, to then **press** `ENTER` twice. Or, **click** ▼ to view a list of Languages and ***double-click*** your selection.

➢ The image displays in the Presentation Language selected (see *Figure 26).* **Repeat** the process to change the display back to the Design Language.
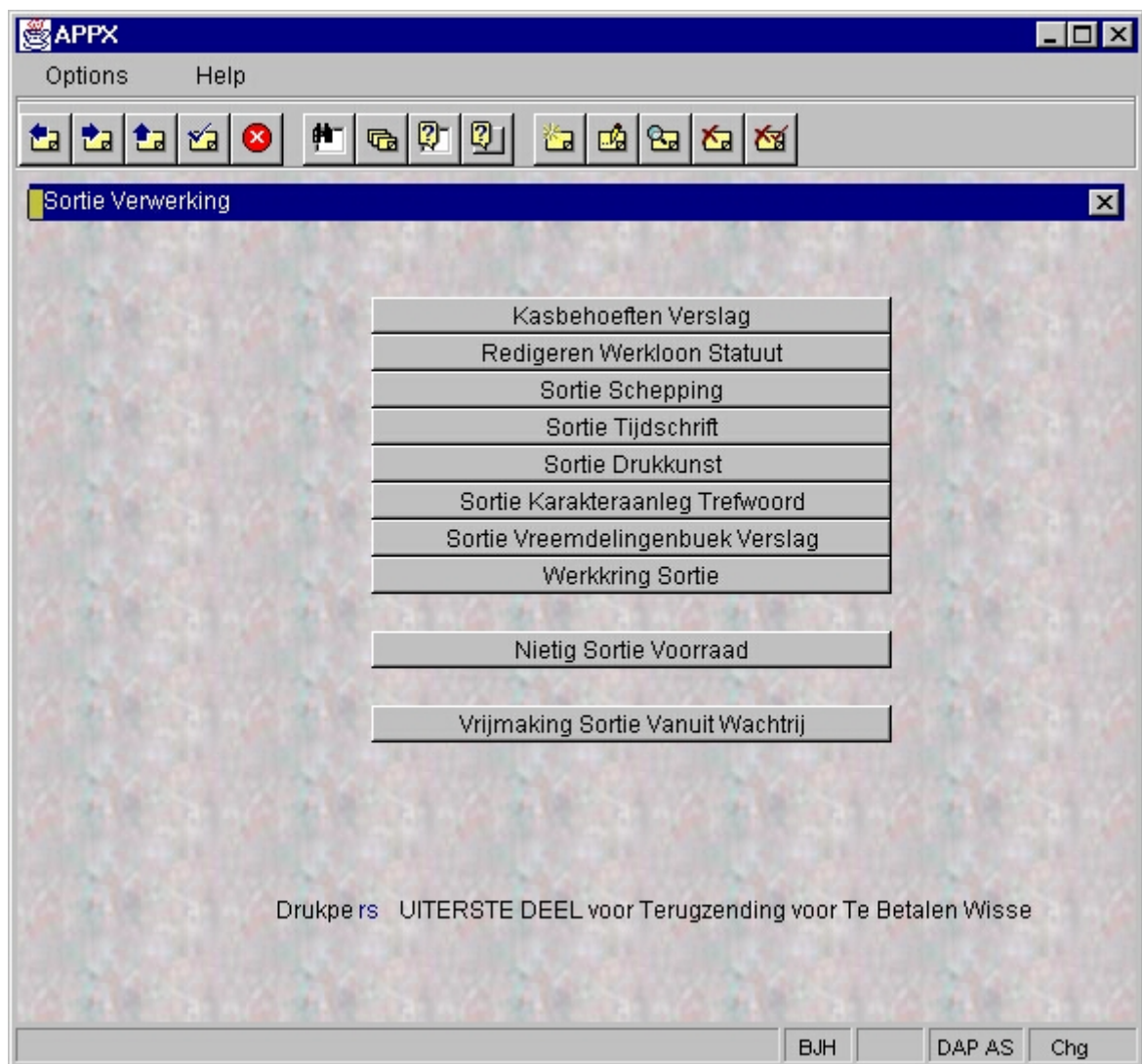

Figure 26 – Previewing A Translation From Application Design

OPTION 95: EDIT/DEFINE TRANSLATION EXCEPTIONS

This option is detailed in the section *The Exception File*.

appx SOFTWARE

TROUBLESHOOTING
FEHLERSUCHE
LOCALIZACIÓN Y SOLUCIÓN DE PROBLEMAS

If you have any problems using the *APPX Language Translator*, reread the appropriate sections of this guide paying special attention to the NOTES, TIPS, and WARNINGS. If you are still unable to resolve the problem, visit the Customer Support page of www.appx.com and look for the *Language Translator's FAQ document*. This document is updated regularly with new issues and solutions.

If you require further assistance contact your APPX support provider or APPX Software, Inc. at 1-800-TRY-APPX.