Languages and GRIN-Global

Updated April 27, 2012

Contents

GRIN-Global and Languages

Overview

The default GRIN-Global (GG) software and interface is in English; however, GG was designed to handle an unlimited number of languages. GRIN-Global stores the language information in its database, in the same way that it stores curatorial data. Your organization can work with GG in any language, but you must first import the language data into GG. After this is completed, users can then interact with GG in their preferred language. This section explains what language data must be supplied.

The following languages are included in Version 1.0:

- English
- Spanish
- Russian
- French
- Arabic

The language used for the interface is distinct from the language used to store the curatorial data. For example, you can run the system in English, but store all of your data in Spanish. Conversely, you could store your data in English, but use Spanish for your preferred interface language.

Directions for adding another language to the GG database are described later in the section *Adding a Language to GRIN-Global* on page 2.

Language-related Database Items in GRIN-Global

Initially, if the language to be used is already installed, your organization should review the translations for that language. If necessary, to meet specific organizational needs, the administrator may need to edit some of the language translations manually or even sometimes may need to export an extensive set of language translations from GG and then re-import them back into the GG database after editing them.

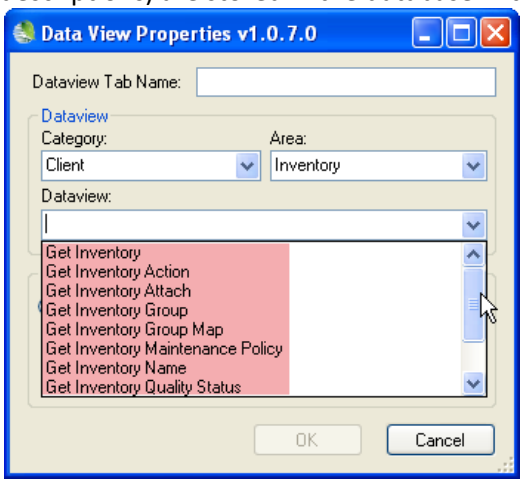
The following language-related items are stored In the GG database:

1. **Table & Dataview Field Name Titles**
   These field name titles are the "language-friendly names" that display as the column headings in the Curator Tool. (These names are stored in two tables: **dbo.sys_table_field_lang** and **dbo.sys_dataview_field_lang**. Detailed information explaining why two tables are used for storing the table field translations is in a subsequent section of this document.)

2. **Application Resources**
   This table stores the "language-friendly text" for the buttons, check boxes, radio buttons, text objects, error messages, labels and titles for all GG user applications such as the Curator Tool or the Public Website, so that these applications can display their interface in any language. (These interface objects are stored in the **dbo.app_resource** table.)

3. **Titles and Descriptions for Dataviews**

   In the Curator Tool, a user can select a dataview from a list when adding a tab for that dataview to the grid. The titles are displayed in the user's currently-selected language. (These titles (and descriptions) are stored in the database in the **dbo.sys_dataview_lang** table.)



4. **Group Names & Code Values**

   Some GRIN-Global fields restrict the user to selecting from a set of possible values. These code values are stored in the **dbo.code_value** and **dbo.code_value_lang** tables. After editing, when a curatorial record is stored, the value selected from the code table is stored in the record's field.

   For example, the inventory record has a **Pathogen Status** field. There are four possible statuses: **FREE, INFECTED, TESTED**, or **UNTESTED**. If the user completes this field, one of these four values must be selected.

Adding a Language to GRIN-Global

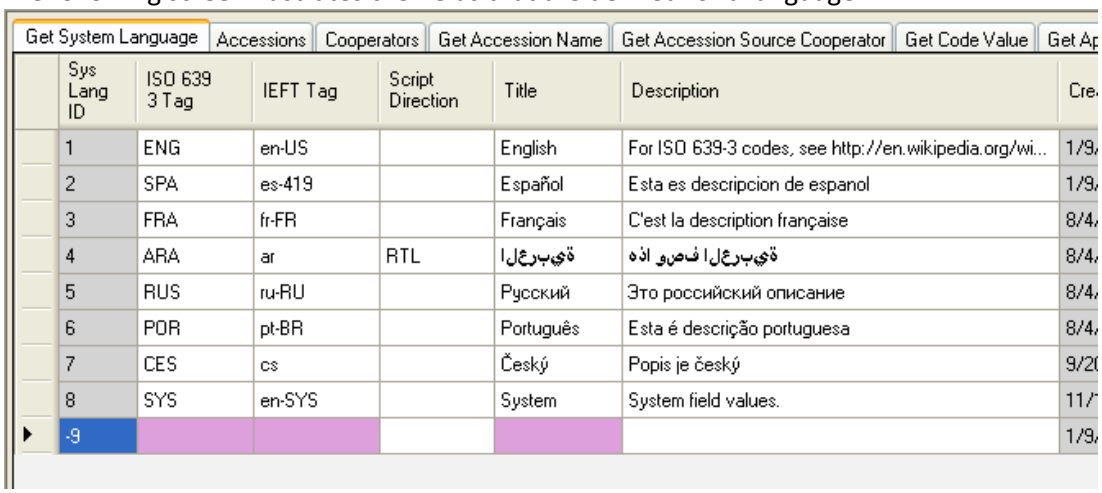
Overview

The five installed languages included in the GG 1.0 database are listed on page 2. If one of these is the language that you intend to run GG under, then this section can be skipped. (There is no compelling reason to remove the "extra" languages. Even if never used, they do not impact run time or create any additional overhead.)

If a language that is intended to be used is not installed, then the organization's administrator will add the language to the GG **sys_lang** table via the **get_system_language** dataview.

The System Language Dataview

The following screen ilustrates the fields that are defined for a language:



Get System Language | Accessions | Cooperators | Get Accession Name | Get Accession Source Cooperator | Get Code Value | Get Ap

Sys Lang ID	ISO 639 3 Tag	IEFT Tag	Script Direction	Title	Description	Cre
1	ENG	en-US		English	For ISO 639-3 codes, see http://en.wikipedia.org/wi...	1/9/
2	SPA	es-419		Español	Esta es descripcion de espanol	1/9/
3	FRA	fr-FR		Français	C'est la description française	8/4/
4	ARA	ar	RTL	ةيبرعلا فصو اذه	ةيبرعلا	8/4/
5	RUS	ru-RU		Русский	Это российский описание	8/4/
6	POR	pt-BR		Português	Esta é descrição portuguesa	8/4/
7	CES	cs		Český	Popis je český	9/2(
8	SYS	en-SYS		System	System field values.	11/7
-9						1/9/

To include a new language in GRIN-Global, in the Curator Tool open the **Get System Language** dataview and add a new record. (The dataview fields are described below.)

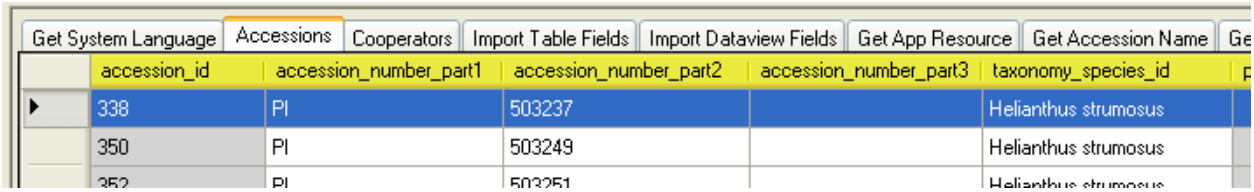
From the Data Dictionary

Field	Description
ISO 639 3 Tag	The ISO 639-3 representation for the language. e.g. "ENG" See http://en.wikipedia.org/wiki/ISO_639-3 and http://www.sil.org/iso639-3/codes.asp?
IEFT Tag	The value in this field must be unique for every language row in the table. The value in this field represents the IETF language tag for the language. e.g. "en-US" See http://en.wikipedia.org/wiki/IETF_language_tag
Script Direction	Direction in which the language is displayed. "RTL" means "right to left." All other values are interpreted as "left to right."
Title	A human-friendly title for the language, e.g. "English"
Description	A human-friendly description for the language, e.g. "United States English"

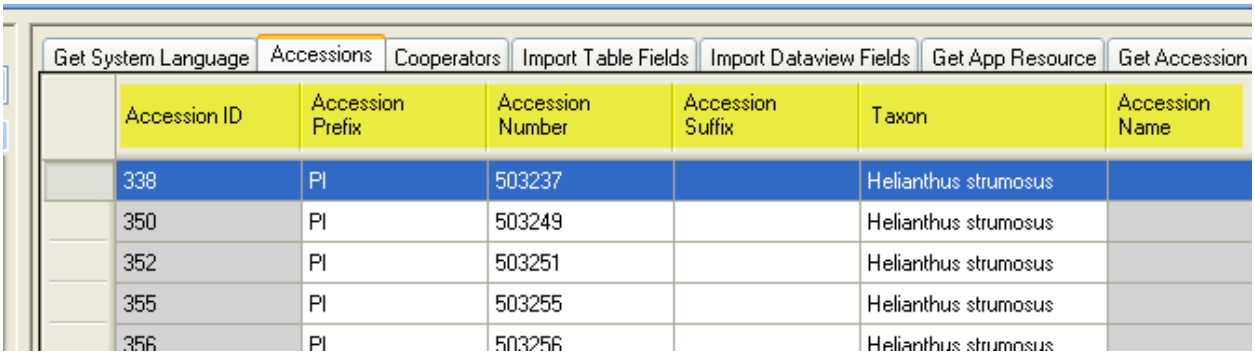
Table & Dataview Field Names

Overview

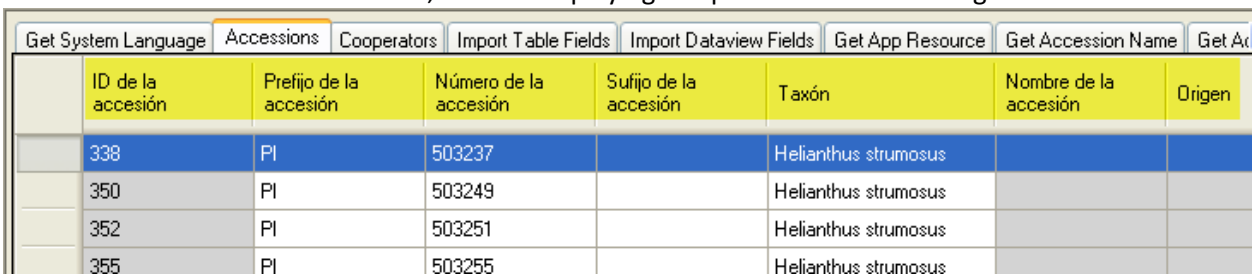
The actual database names for the column headings are more technical than friendly; fortunately the Curator Tool user doesn't normally see these:



Instead, the CT user sees "language-friendly names" displayed. Some of the English friendly names for the **Accessions** dataview are shown here:



This is the same Accession dataview, but it is displaying its Spanish column heading names:

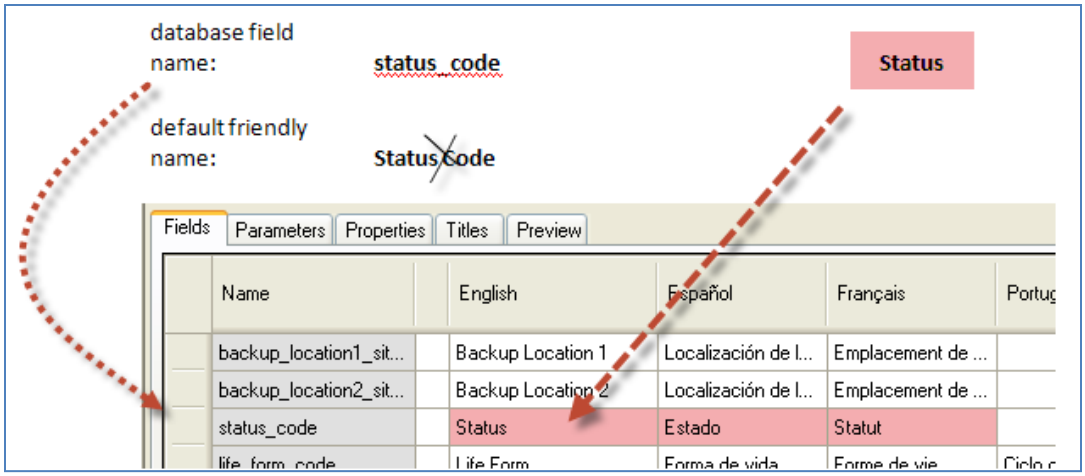


These column headings in the Curator Tool are modifiable since they are stored as system data in the GG database. (This data is referred to as *system* data in contrast to the curatorial data that is used for accession management.)

Storing Language-Friendly Column Heading Names at Two Levels (Overview)

Organizations can decide what specific text will be displayed for each column heading.

For example, in the main accession dataview, when viewing in English, the user will see "**Status**" as the heading for the **status_code** field ("**status_code**" is the actual database field name.) For each language, "**status_code**" is replaced with a language-specific name. In Spanish, the language-friendly translation is "**Estado,**" and in French it is "**Statut**."



If an organization has added a new language to GRIN-Global, then all of the language-friendly, column-heading names for that language also need to be added to the database by the administrator.

As background, "friendly" heading names can be defined at two levels:

- *table*/field level
- *dataview*/field level

The table/field friendly heading name serves as the default heading name whenever a new dataview is created. For example, any new dataview that points to **accession.accession_id** will automatically receive "Accession ID" for its English friendly name.

However, an administrator editing a dataview can override a default heading by modifying it in the Admin Tool's dataview editor. For example, "Accession ID" is the default heading for the **accession.accession_id** field. But in one particular dataview it might be desirable to have "Accession" instead. To override the default "Accession ID," the administrator editing the dataview would simply replace the text "Accession ID" in the dataview editor with "Accession." When the change to the dataview is saved to the database, the translation text "Accession" is saved to the database at the dataview/field level.

Note that the value of the heading stored at the *table*/field level is left unchanged –only the value of the heading stored at the *dataview*/field level is changed. All the other dataviews pointing to **accession.accession_id** will continue to display the default heading name "Accession ID."

Why the apparent duplication? The table/field friendly name is a fallback if nothing is specified at the dataview/field level. Almost all of the GRIN-Global dataviews currently use table/field as the friendly name source. One important reason for using the table/field level translations as the primary translation storage place is this: if a change is made to the translation of a heading at this level, it will flow down to all of the dataviews using the default heading translation –none of those dataviews will need to be modified.

Changing a Dataview's Language-friendly Column Heading Names

There are two very different approaches for changing the column-heading names;

- change the entire set of default heading names at one time
  (updating the defaults heading names stored in the **dbo.sys_*table*_field_lang** table)

- edit individual heading names for specific dataviews (thus overriding the default friendly names; updating the **dbo.sys_*dataview*_field_lang** table)

Changing the Default Language-friendly Column Heading Names

The pair of dataviews with "table" in their name is used to manage the default language-friendly column heading names. The other pair, with "dataview" in their name, is used to override the default names.

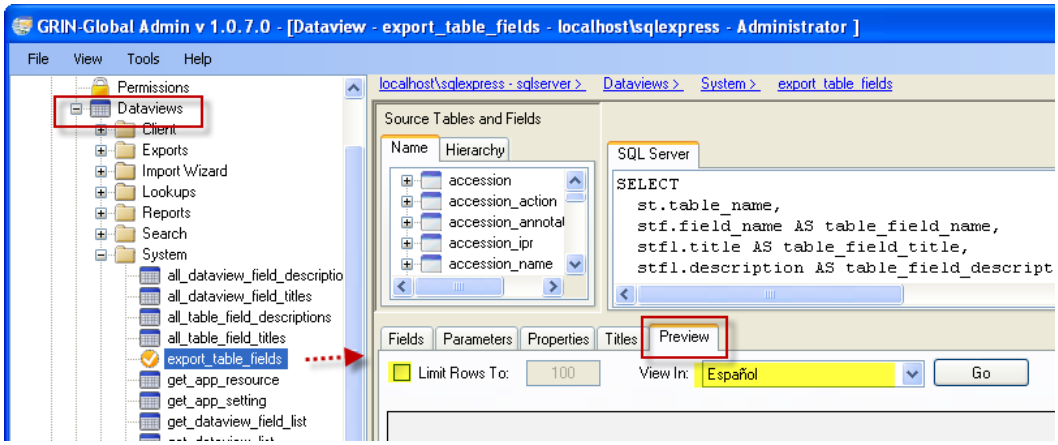
Dataview	Description
export_table_fields	Export the default, language-friendly column headings from GG into a spreadsheet. Also provides a side-by-side comparison of the English names with the language selected when the dataview is invoked. These default dataview heading names are stored in the **dbo.sys_table_field_lang** table.
import_table_fields	An Import Wizard dataview for importing the data back into GG.
export_dataview_fields	Use this dataview to review and edit the language-friendly column heading names that have been modified.
import_dataview_fields	This dataview is accessed via the Import Wizard to import the data back into GG. It updates the data stored in the **dbo.sys_dataview_field_lang** table.

Exporting the Default Friendly Column Headings from GG into a Spreadsheet

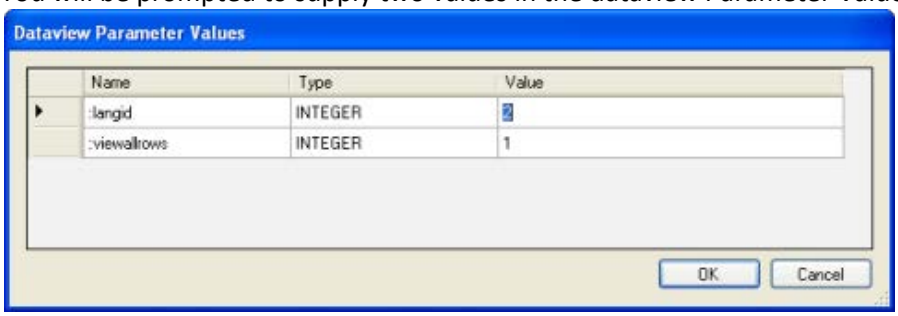
export_table_fields

In the Admin Tool, open the **export_table_fields** dataview; in the editor window, switch to **Preview** tab.

Deselect (remove the check) in the **Limit Rows to 100** checkbox. In the **View In** box, select the language to be used – generally this language will be the same as the language of the records you are exporting; click **Go**.



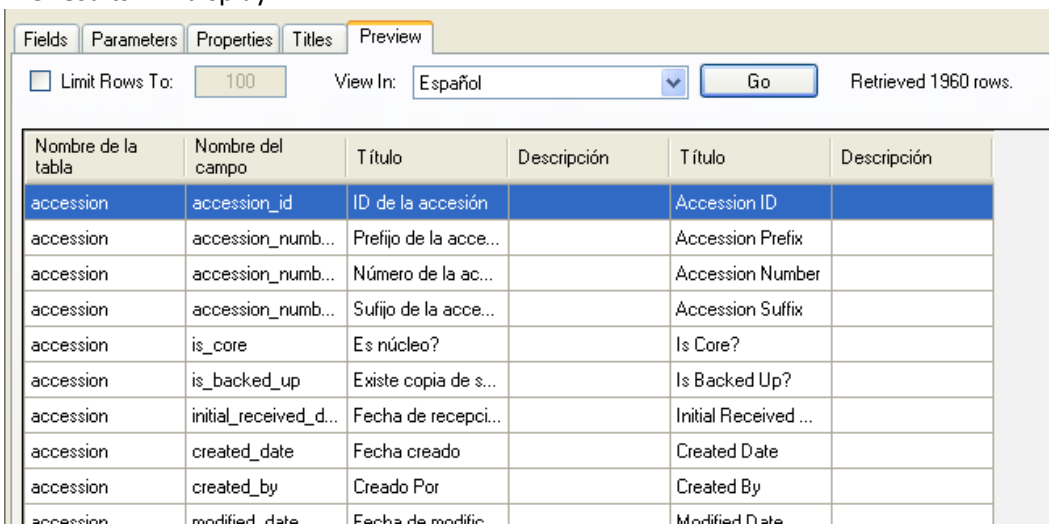
You will be prompted to supply two values in the dataview Parameter Values window:



The first row is looking for the **Sys Lang(uage) ID** value – see page 2 or the table below. In the example, "2" was entered for the first parameter, indicating Spanish (Español); for the second parameter always input "1"– this indicates that all rows will be selected.

Sys Lang ID	Title
1	English
2	Español
3	Français
4	العربية
5	Русский
6	Português
7	Český
8	System

The results will display:



This **export_table_fields** dataview was designed to list the field names for the specified language as well as the English friendly names for comparison. Currently, in 1.0, the **Description** field is not being used and can be ignored.

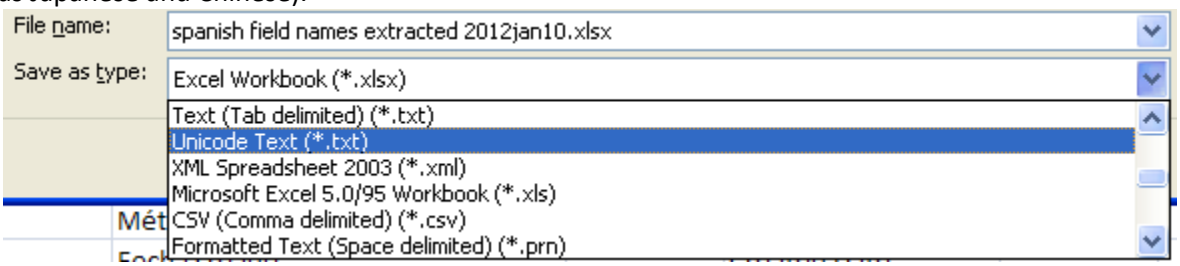
Select all the records and copy into an Excel spreadsheet. (In an English keyboard, the Windows shortcut keys **Ctrl-A** will select all rows and **Ctrl-C** will copy them to the clipboard.)

Results pasted into an Excel spreadsheet:

	A	B	C	D	E	F
1	**Nombre de la tabla**	**Nombre del campo**	**Título**	**Descripción**	**Título**	**Descripción**
2	accession	accession_id	ID de la accesión		Accession ID	
3	accession	accession_number_part1	Prefijo de la accesión		Accession Prefix	
4	accession	accession_number_part2	Número de la accesión		Accession Number	
5	accession	accession_number_part3	Sufijo de la accesión		Accession Suffix	
6	accession	is_core	Es núcleo?		Is Core?	
7	accession	is_backed_up	Existe copia de seguridad?		Is Backed Up?	

In Excel, edit the **Titles** in column **C** and save your changes. (In the example shown here, the titles are the Spanish (Español) titles.)

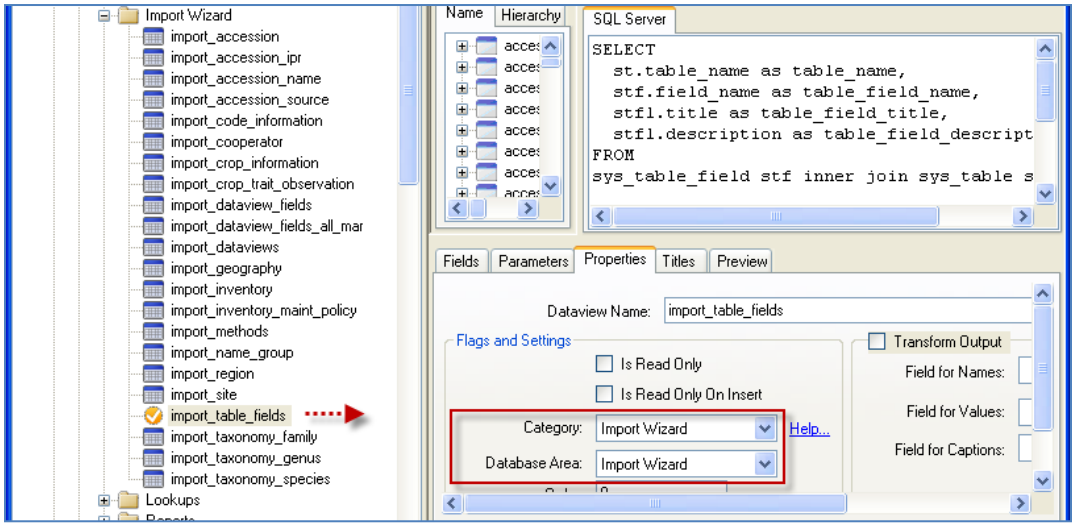
When saving, use the **Unicode Text (*.txt)** format as shown below. This choice will automatically save the file as tab-separated-values with Unicode encoding (which is needed for double-byte languages such as Japanese and Chinese).



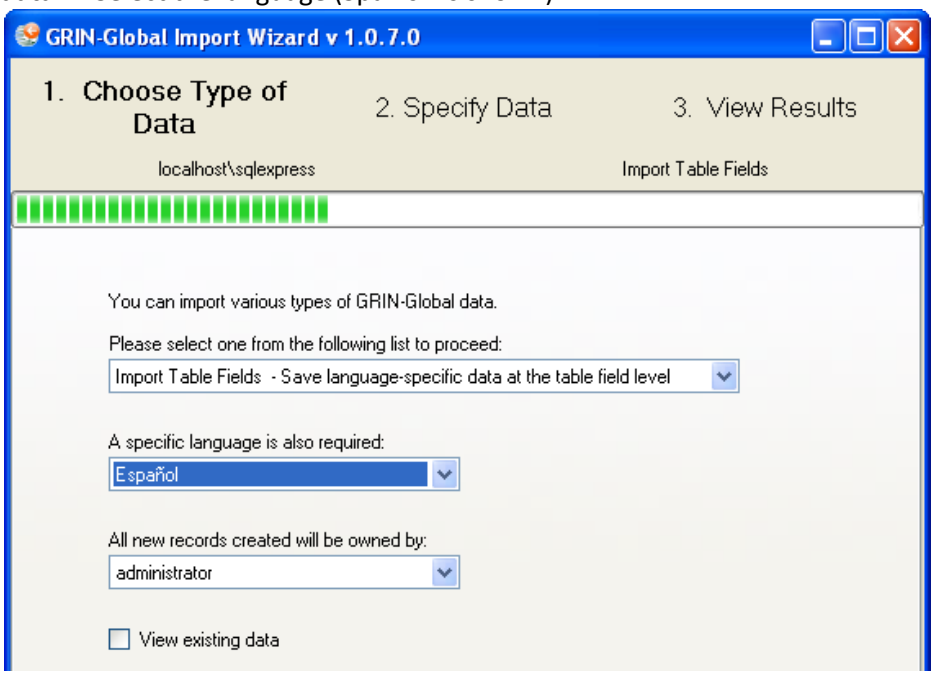
import_table_fields

To import the data back into GG, in the Admin Tool, use the Import Wizard dataview (**import_table_fields**) that has been designed for that purpose.

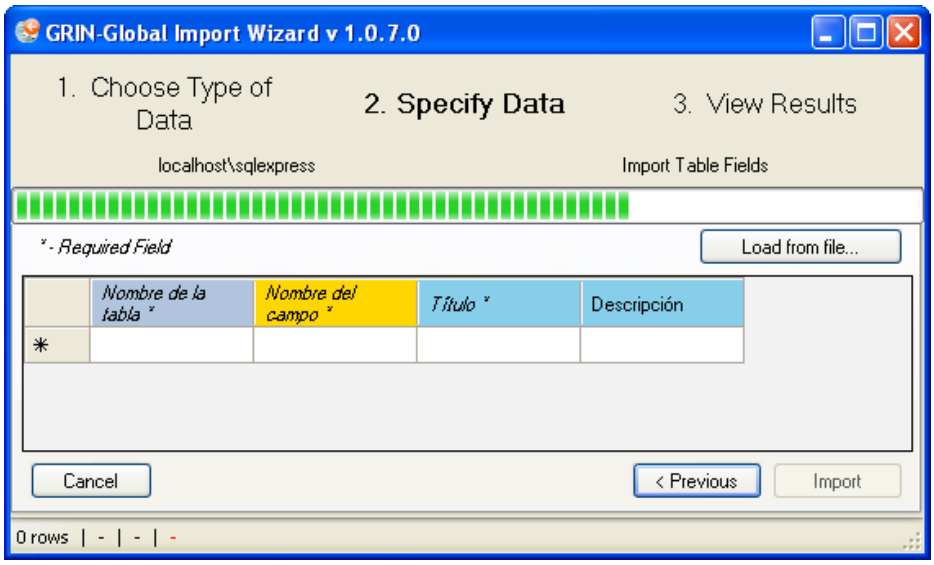
Note that when GG is initially installed, this dataview might be listed under **System** and not **Import Wizard**. If that is the case, change the **Category** and **Database Area** properties for the **import_table_fields** dataview.



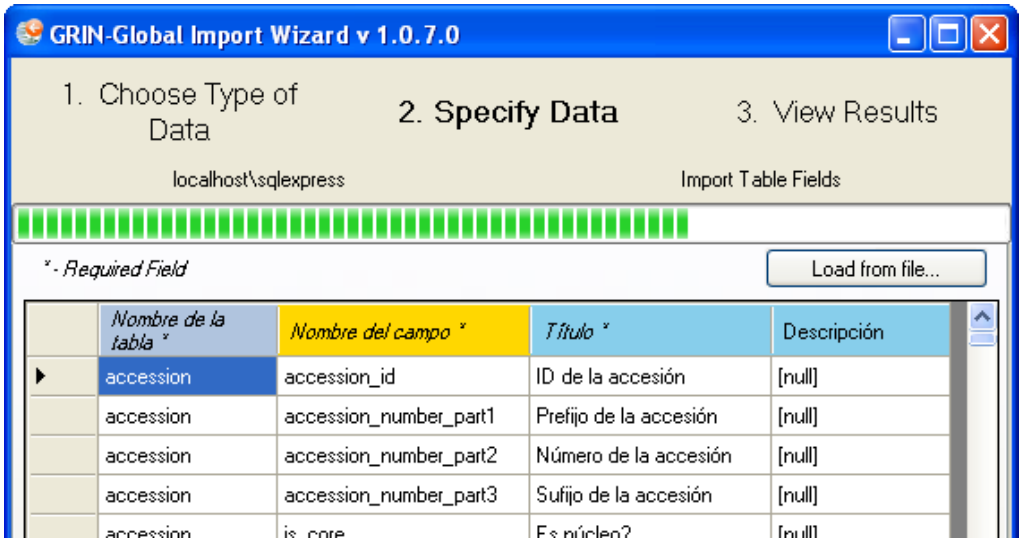
In the Admin Tool, start the Import Wizard and select "**Import Table Fields – Save Language-specific data…**" Select the language (Spanish is shown):



In Excel, copy the first four columns, A through D, from your spreadsheet that was discussed earlier in the section *export_table_fields* on page 9. Then paste the data into the grid in the Import Wizard (alternatively, click the **Load from file...** button and import the .txt file you created); click the **Import** button.



Results:



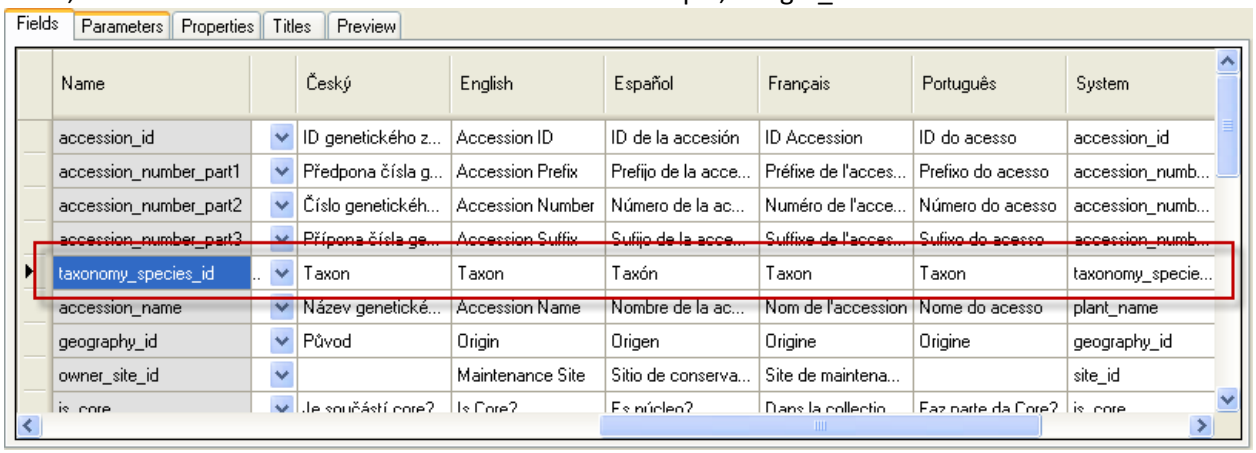
There is no need to export/import the entire set of records for a language if you just intend to change one or a few rows. You can export them all, and then in a spreadsheet, selectively choose and edit the records that you intend to modify. Import only the modified records, but remember to include the header row (to correspond with the IW's headings.)

By completing the instructions above, you have successfully replaced the default, language-friendly heading names in the **dbo.sys_table_field_lang table.**

Overriding a Dataview's Default Column Heading Names

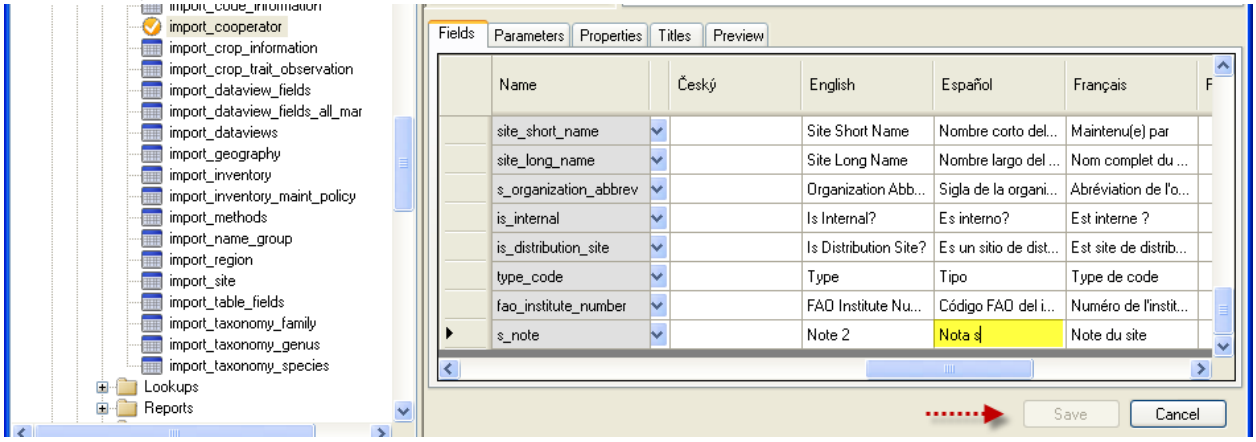
Manually / individually

To edit a single dataview's language-friendly name and to override the default language-friendly heading name, use the Admin Tool's dataview editor. As an example, the get_accession dataview is shown:



If for some reason your organization did not want “Taxon” as the heading, the administrator could edit the heading by using the dataview editor in the Admin and modify the text used for the heading. Note that the dataview editor displays the friendly names for every installed language.

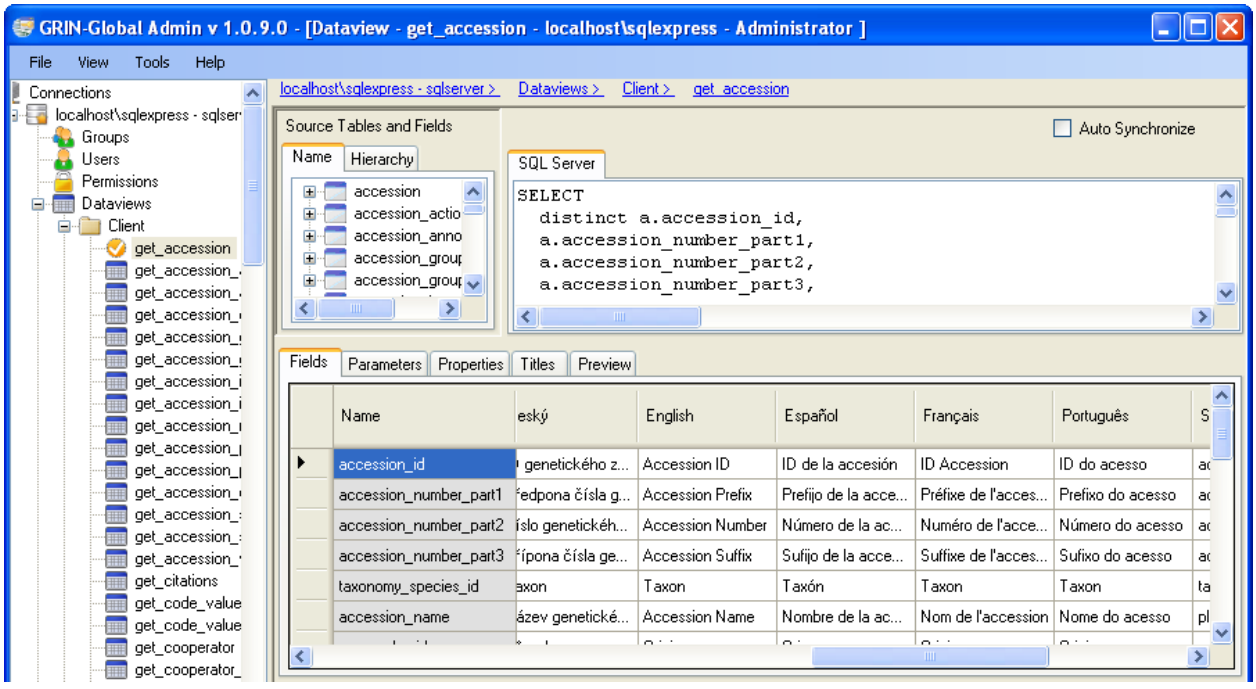
Input the new heading text under the respective language column; click the **Save** button. In the example shown here, the



Note that manually editing a dataview’s language-friendly column heading name in the dataview editor as explained above does not change the heading of the same field when it is used in other dataviews. If you want to change the table field’s language-friendly heading-name for all dataviews, follow the directions described in the section beginning on page 7.

Reviewing the Entries in the dbo.sys_dataview_field_lang Table

The default language-friendly column names are stored in the **dbo.sys_table_field_lang** table. We mentioned that the default language-friendly column names can be manually overridden by the administrator using the Admin Tool’s dataview editor. For example, shown is the **get_accession** dataview:



When friendly names of specific columns are manually edited, these names are stored in the **dbo.sys_*dataview*_field_lang** table.

In a new installation, it may be desirable for the organization's administrator to review all existing entries in the **dbo.sys_dataview_field_lang** table that were provided in the installation. The following directions explain how to review the table.

export_dataview_fields

Use the **export_dataview_fields** dataview to review and edit the language-friendly column heading names that have been modified.

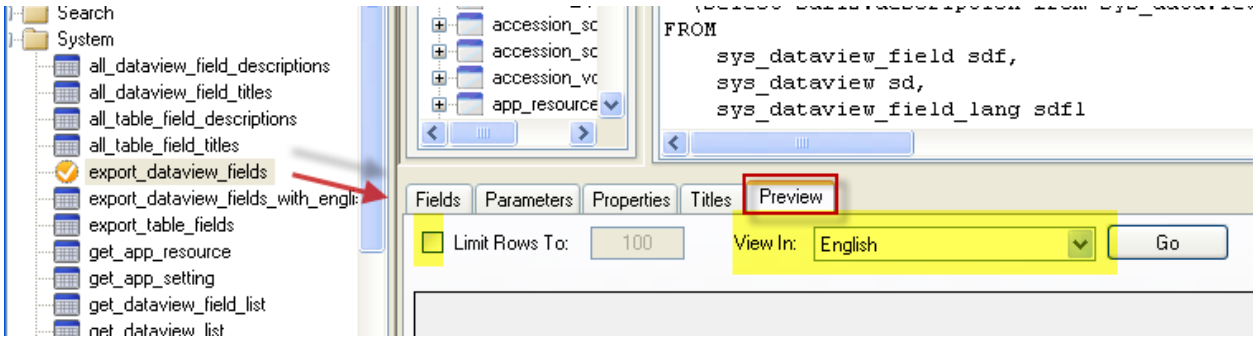
The directions for exporting and then importing back into GG the dataview fields follow the same flow as the directions for exporting and importing back the table field names that were described on page 7, except that you use the **export_dataview_fields** and **import_dataview_fields** dataviews.

Note: the spreadsheet file will have fewer records than the spreadsheet created when the table field names were exported because there are not many dataviews overriding the default column heading names.

Detailed Steps

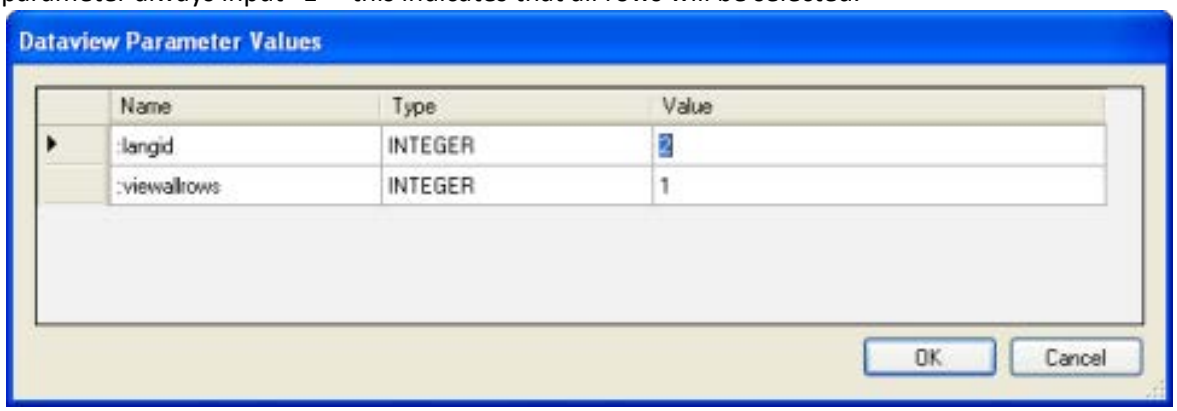
In the Admin Tool, open the **export_dataview_fields** dataview; in the editor window, switch to **Preview** tab.

Deselect (remove the check) in the **Limit Rows to 100** checkbox. In the **View In** box, select the language to be used – generally this language will be the same as the language of the records you are exporting; click **Go**.



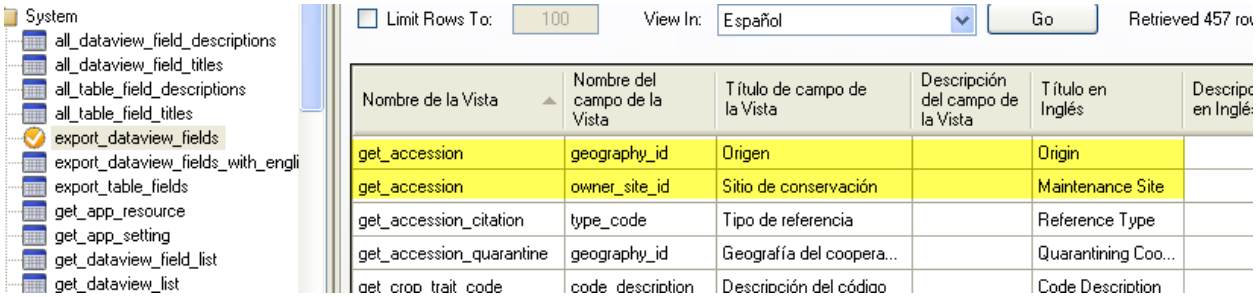
You will be prompted to supply two values in the dataview Parameter Values window. The first row is looking for the **Sys Language ID** value – see page 2 or the table below the following graphic. In the example, "2" was entered for the first parameter, indicating Spanish (Español); for the second

parameter always input “1”– this indicates that all rows will be selected.



Sys Lang ID	Title
1	English
2	Español
3	Français
4	
5	Русский
6	Português
7	Český
8	System

In the following example, the Spanish entries are displayed. Highlighted are the two language-friendly column names that had been overridden for the get_accession dataview:

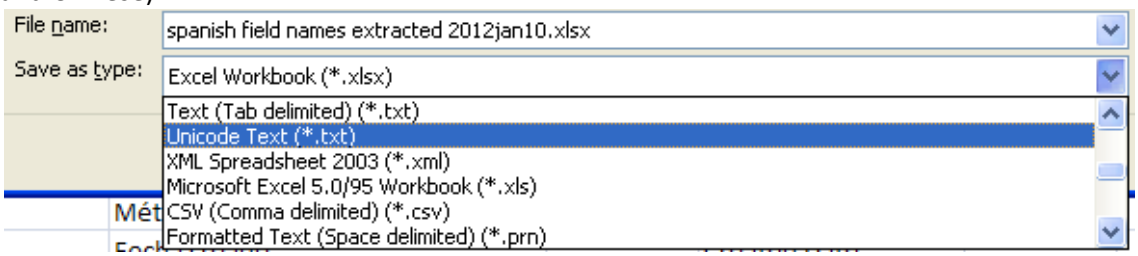


In 1.0, currently the **Description** field is not being used and can be ignored.

Select all the records and copy into an Excel spreadsheet. (In an English keyboard, the Windows shortcut keys **Ctrl-A** will select all rows and **Ctrl-C** will copy them to the clipboard.)

	A	B	C	D	E	F
1	Nombre de la Vista	Nombre del campo de la Vista	Título de campo de la Vista	Descripción del campo de la Vista	Título en Inglés	Descripción en Inglés
2	accession_lookup	value_member	Valor miembro		Value Member	
3	accession_lookup	display_member	Miebro desplegado		Display Member	
4	accession_name_lookup	value_member	Valor miembro		Value Member	
5	accession_name_lookup	display_member	Miebro desplegado		Display Member	
6	accession_source_lo				Display	

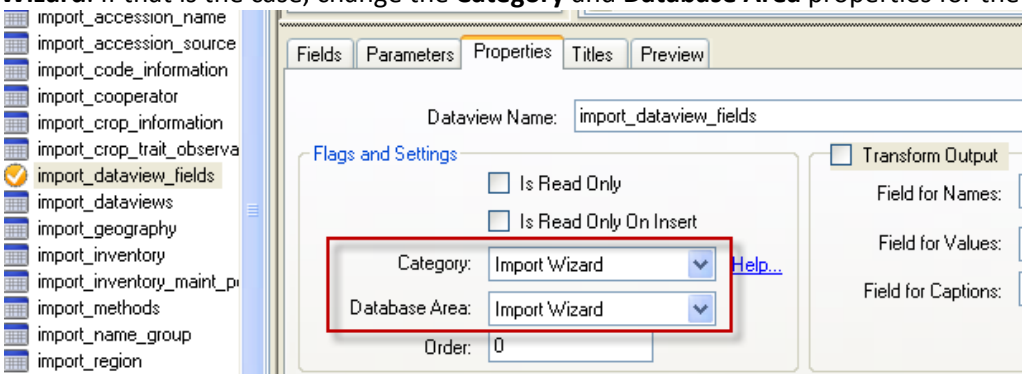
In Excel, edit the **Titles** (Column C) in the language and save your changes. When saving, use the **Unicode Text (*.txt)** format as shown below. This choice will automatically save the file as tab-separated-values with Unicode encoding (which is needed for double-byte languages such as Japanese and Chinese).



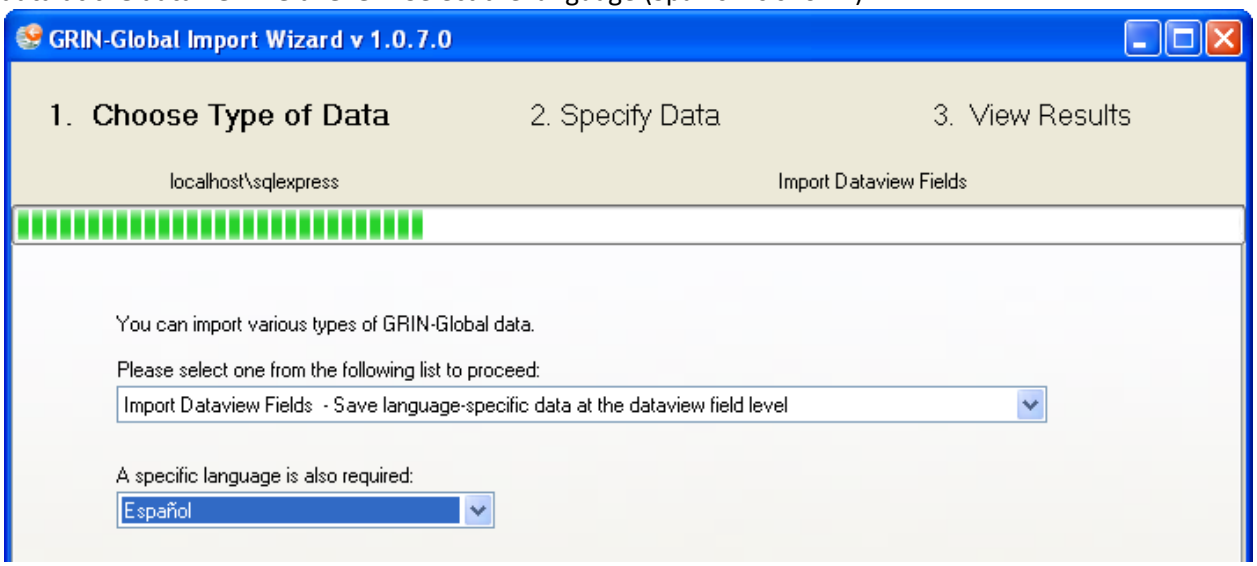
import_dataview_fields

To import the data back into GG, in the Admin Tool, use the Import Wizard dataview (**import_dataview_fields**) that has been designed for this purpose.

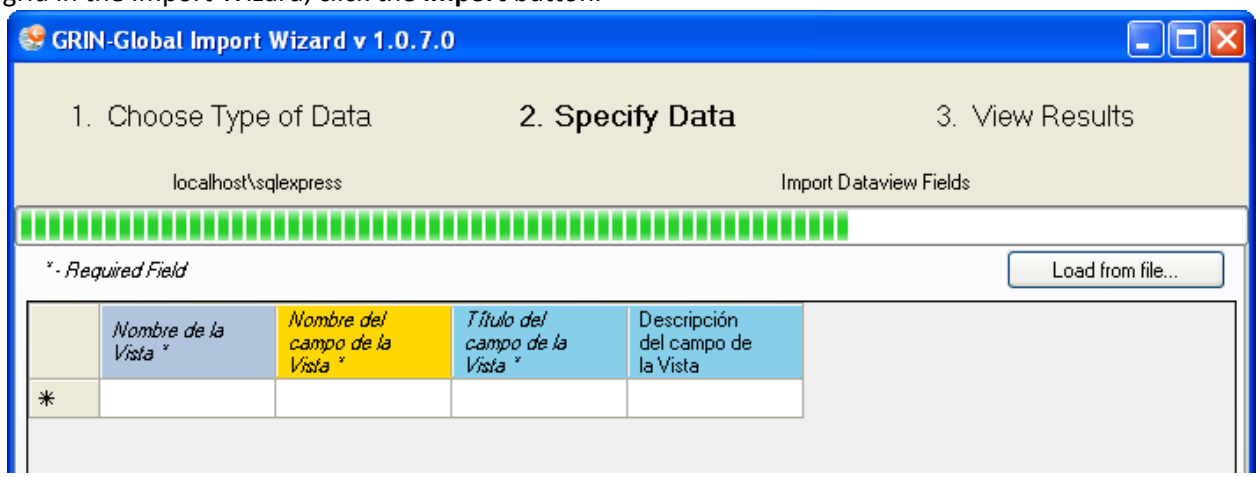
Note that when GG is initially installed, this dataview might be listed under **System** and not **Import Wizard**. If that is the case, change the **Category** and **Database Area** properties for the dataview.



In the Admin Tool, start the Import Wizard and select "**Import Dataview Fields – Save Language-specific data at the dataview field level**." Select the language (Spanish is shown):

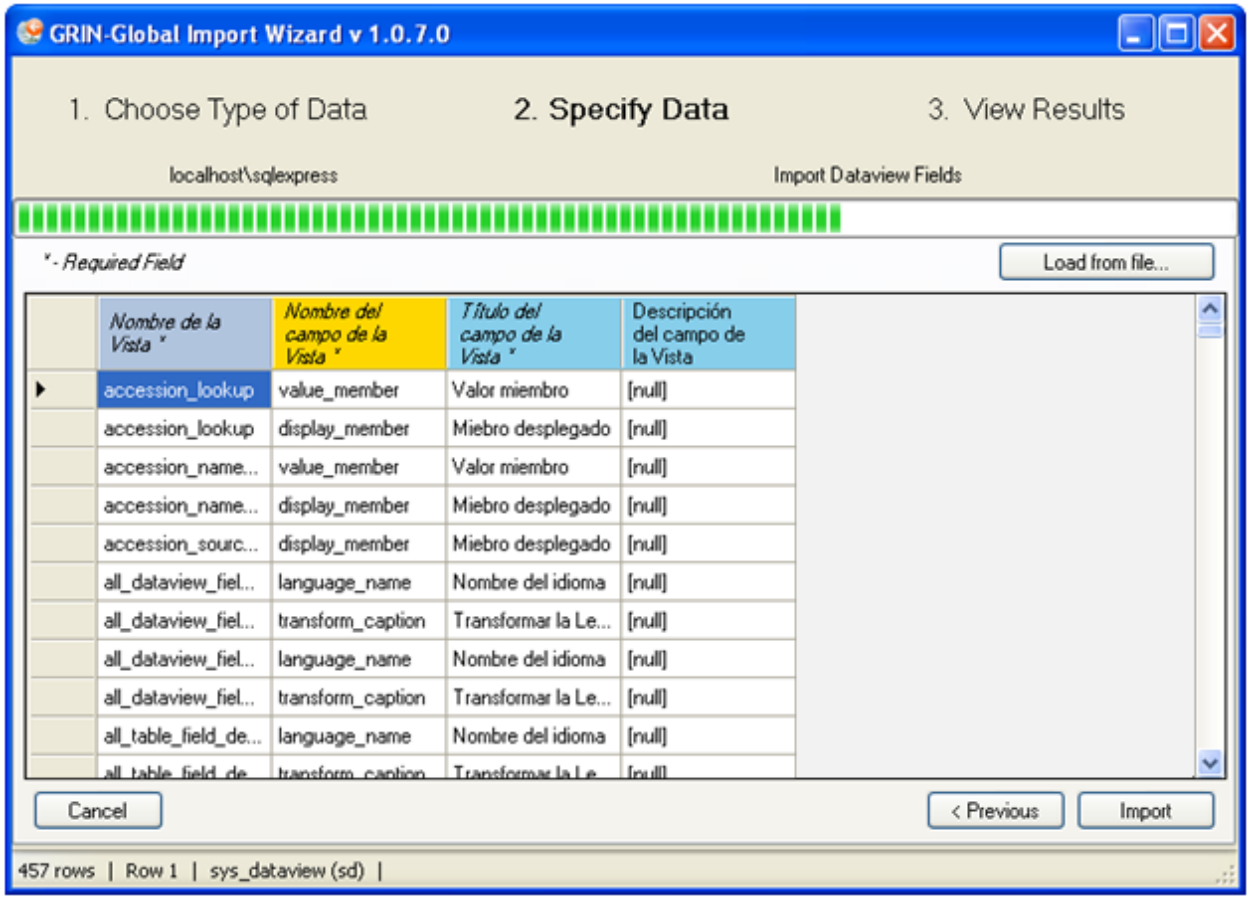


In Excel, copy the first four columns, A through D, from your spreadsheet; then paste the data into the grid in the Import Wizard; click the **Import** button.



As with all Import Wizard imports, the spreadsheet headings must match exactly with the Import Wizard headings.

Results:



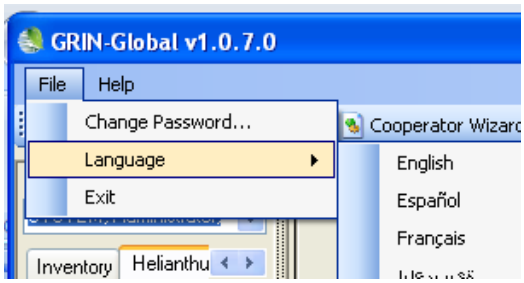
Note

Whenever a dataview has a friendly name added or edited, a record is added to the **dbo.sys_dataview_field_lang** table in the database. If for some reason at a later time the friendly name is to be removed from this table, it will be necessary to delete the modified dataview and then import the dataview back into GG. Complete directions for completely deleting and replacing a dataview are included in the Appendix under the section titled: "Deleting a Dataview."

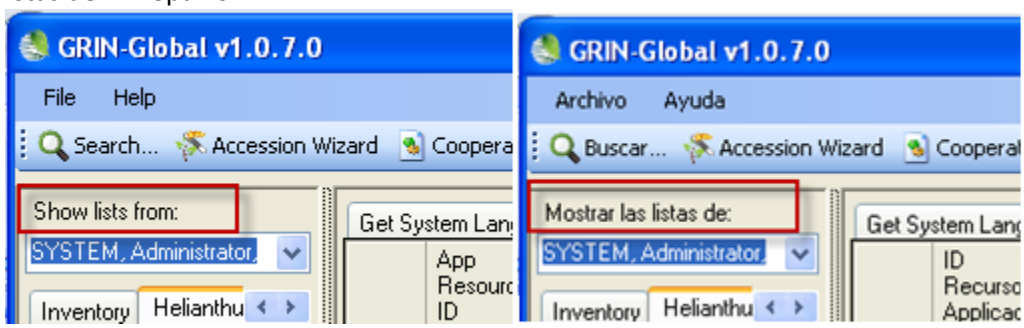
Application Resources

Overview

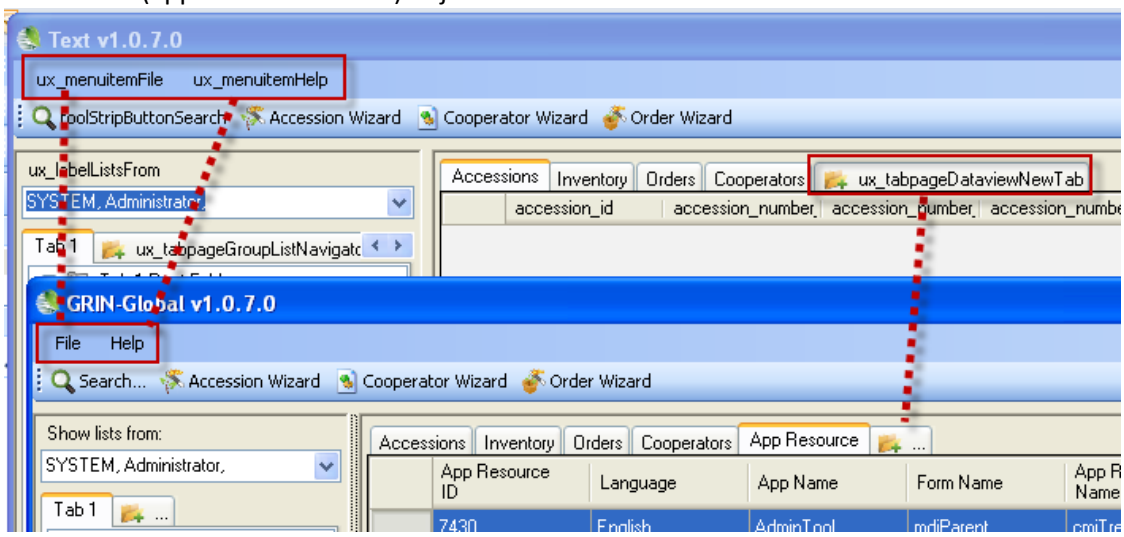
When defining a new user, the administrator also assigns a default language for the user. Throughout the Curator Tool and Admin Tool interface, text items can be displayed in the user's default (ideally his preferred) language. However, in the CT, the user can also select a different language via **File** on the main menu:



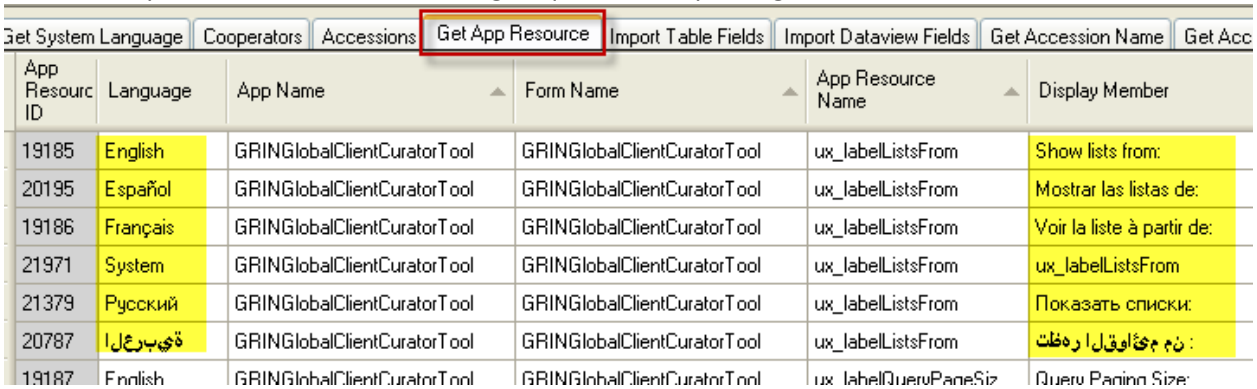
Each language displays its respective labels in the windows, boxes, and buttons. For example, the label above the treeview on the Curator Tool window displays "Show lists from:" in English and "Mostrar las listas de:" in Spanish:



You can have multiple CT windows open – this is very helpful when determining the system name of an (application resource) object:

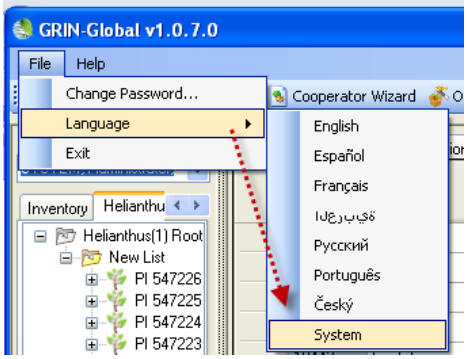


So where are these labels stored? In GRIN-Global's **dbo_app_resource** table. As the administrator, you can edit these text items to meet your organization's needs. The Curator Tool has a dataview in which you can display all of these "application resource" items. You can also use this dataview to copy the text data into a spreadsheet for further editing or quick mass updating.

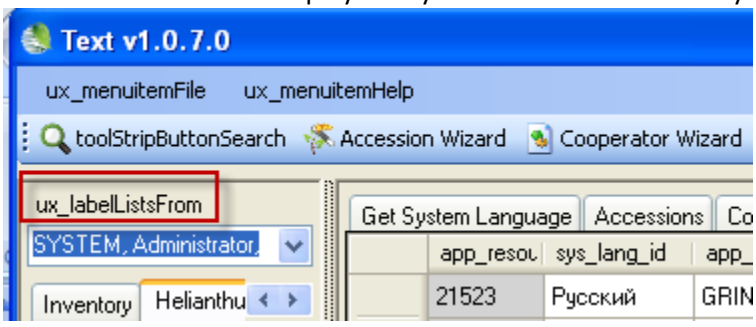


This updating is not generally done often, but is definitely necessary whenever an organization introduces a new language to be used within the GG software.

To determine the item's **App Resource Name**, switch to **System** in the Curator Tool (Select **File** | **Language** | **System**).

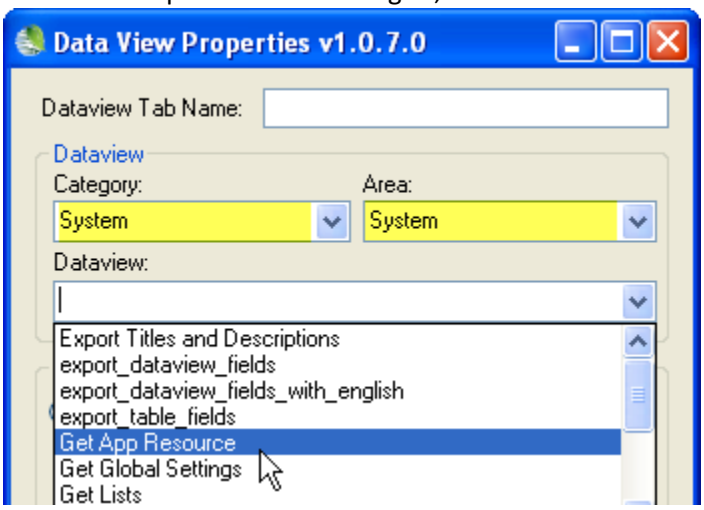


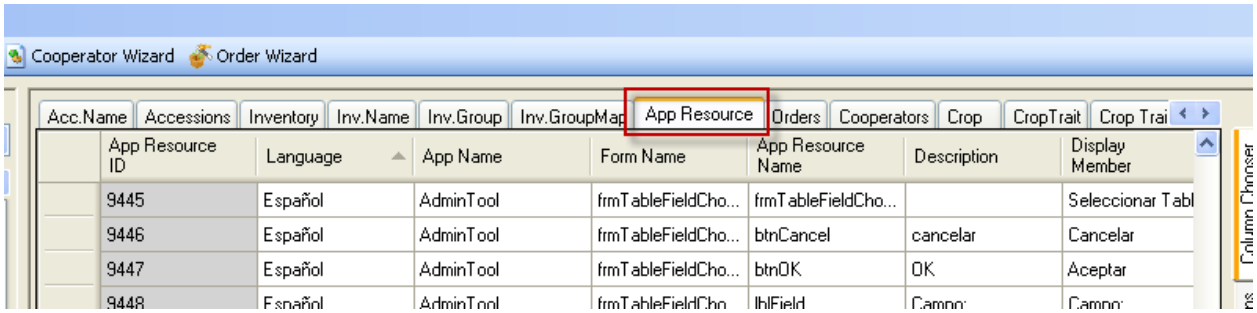
The interface will then display the system names which usually only the software developer uses:



Correcting an Individual AppResource Item

If you need to correct an AppResource item, open the **get_app_resource** dataview in the Curator Tool. Locate the respective row in the grid, and edit and save the record:

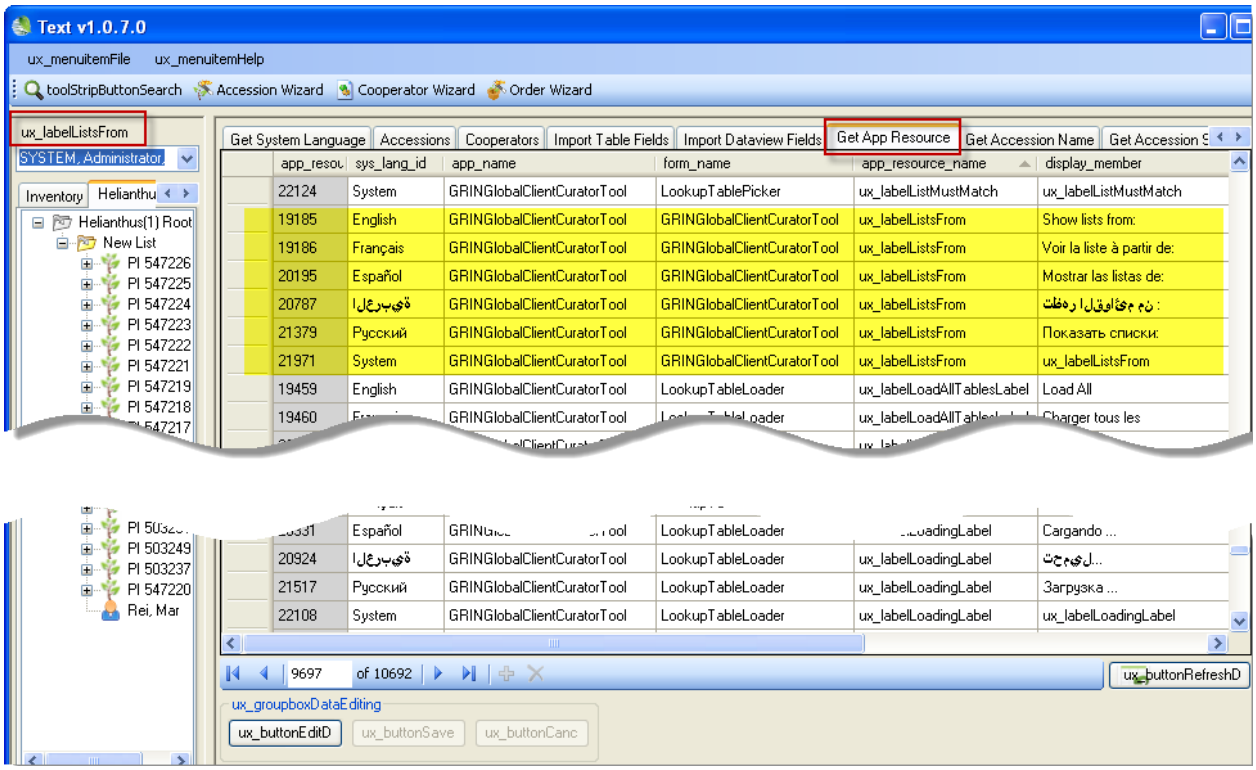




Mass Editing AppResource Items (or Adding Items When Adding a New Language)

Copy the AppResource data from the Curator Tool into a spreadsheet.

When editing existing data, consider sorting the data by app_name, then form_name, app_resource_name, language, etc. Edit the data in the spreadsheet, and then copy back into the CT as you would with any new data.



Adding a New Language's App Resource Records

If you are adding a new language, you could filter the CT for just one existing language, such as English, and then drag those records into Excel.

	A	B	C	D	E	F	G
1							
2	app_resource_id	sys_lang_id	app_name	form_name	app_resource_name	description	display_member
3	7854	English	AdminTool	frmAddSeparator	btnAdd	&Add	&Add
4	7855	English	AdminTool	frmAddSeparator	btnCancel	&Cancel	&Cancel
5	7852	English	AdminTool	frmAddSeparator	frmAddSeparator		Add a new Separator
6	7853	English	AdminTool	frmAddSeparator	lblWord	Character to add as a se	Character to add as a separator:

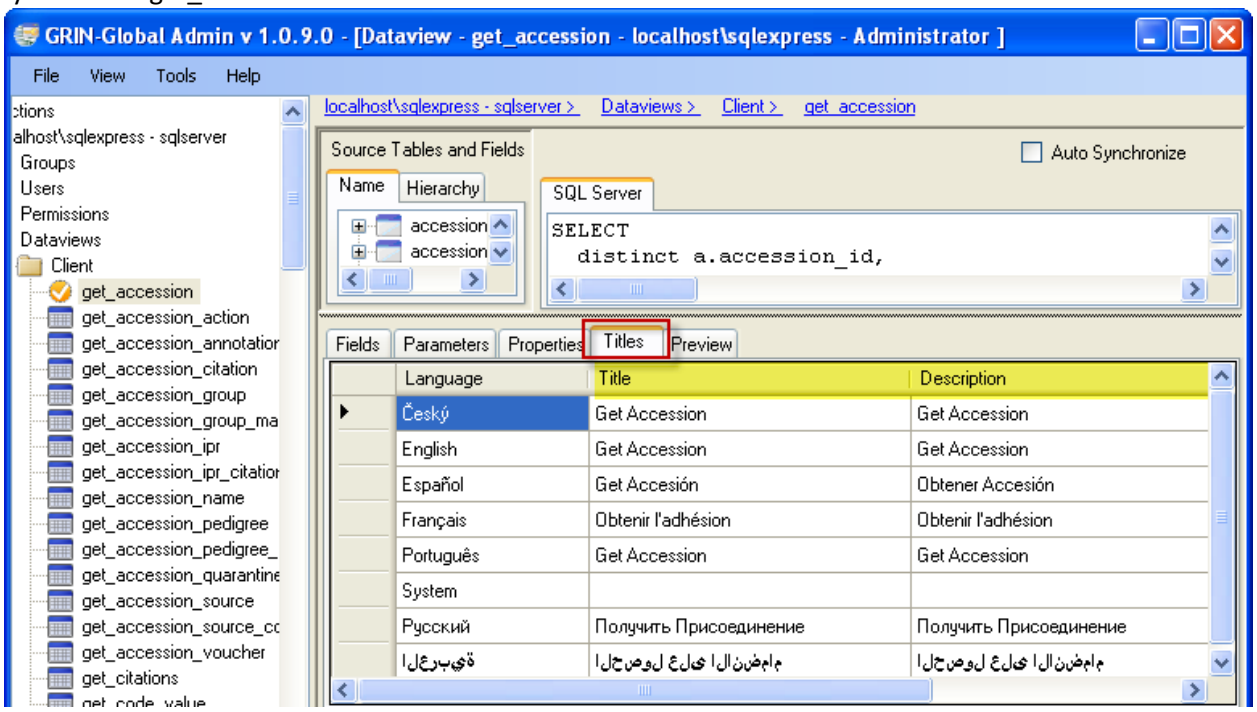
You will need to edit columns **A**, **B**, and **G**. In column...

- A – delete the existing **app_resource_id**'s (since these will be new records and will have their id's assigned by the software)
- B – change the language to match the new language (see p. 4)
- G – using the language, edit the **display_member** fields (It might be easier to insert a new column adjacent to G, input your entries, and then delete the original column G entries, and ensure the heading of the new column is **display_member**

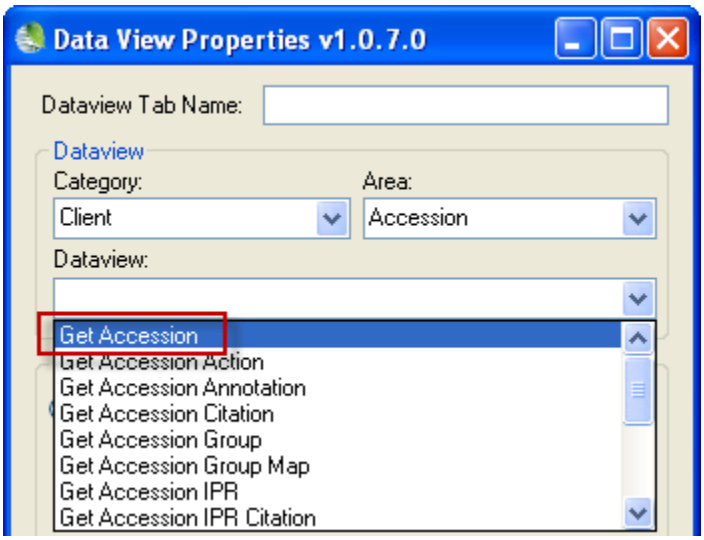
Titles and Descriptions

Overview

Each dataview can be described with a **Title** and **Description** in each of the languages defined in the system. The get_accession dataview is shown here in the Admin Tool:



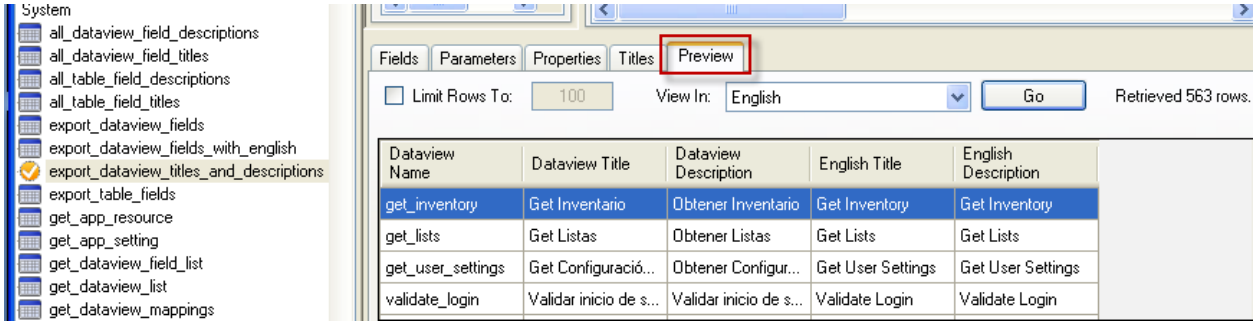
The Curator Tool users see the client dataviews' Titles when they open the **Data View Properties** window:



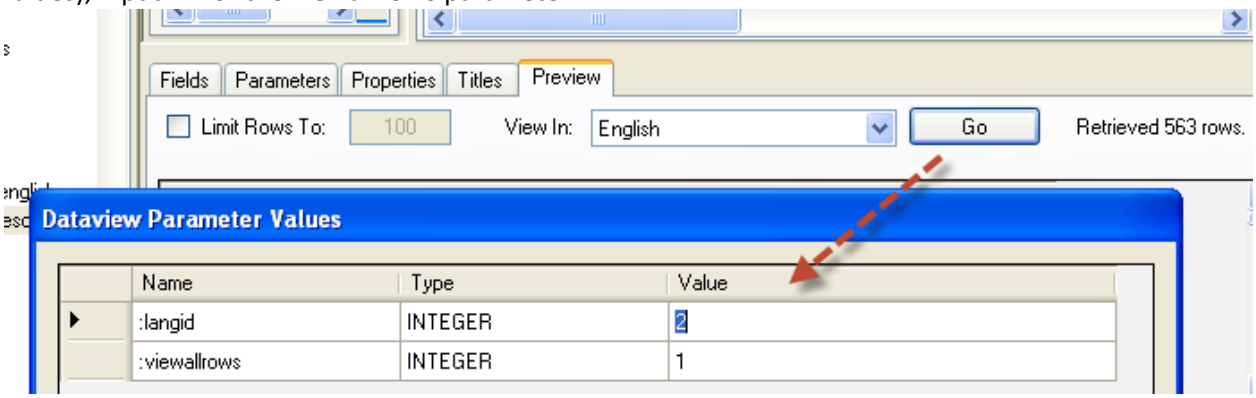
In the Admin Tool dataview editor, you can manually change a dataview's Title and Description. If you need to edit many dataview titles and descriptions, rather than edit them one-by-one, you may prefer to edit them *en masse* using two companion dataviews that can be used to review and edit the Titles and Descriptions. (Directions begin on the next page.)

Exporting Titles and Descriptions

Open the **export_dataview_titles_and_descriptions** dataview. To extract the titles and descriptions; select the **Preview** tab:

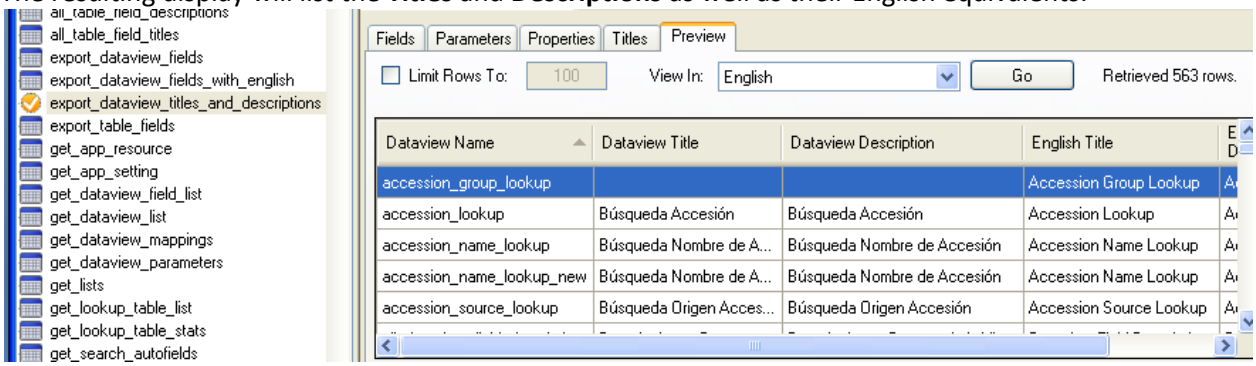


Uncheck the **Limit Rows To:** checkbox; click **Go**; input the value for the desired language (see page 4 for values); input **1** for the **viewallrows** parameter:



"2" indicates Spanish (Español)

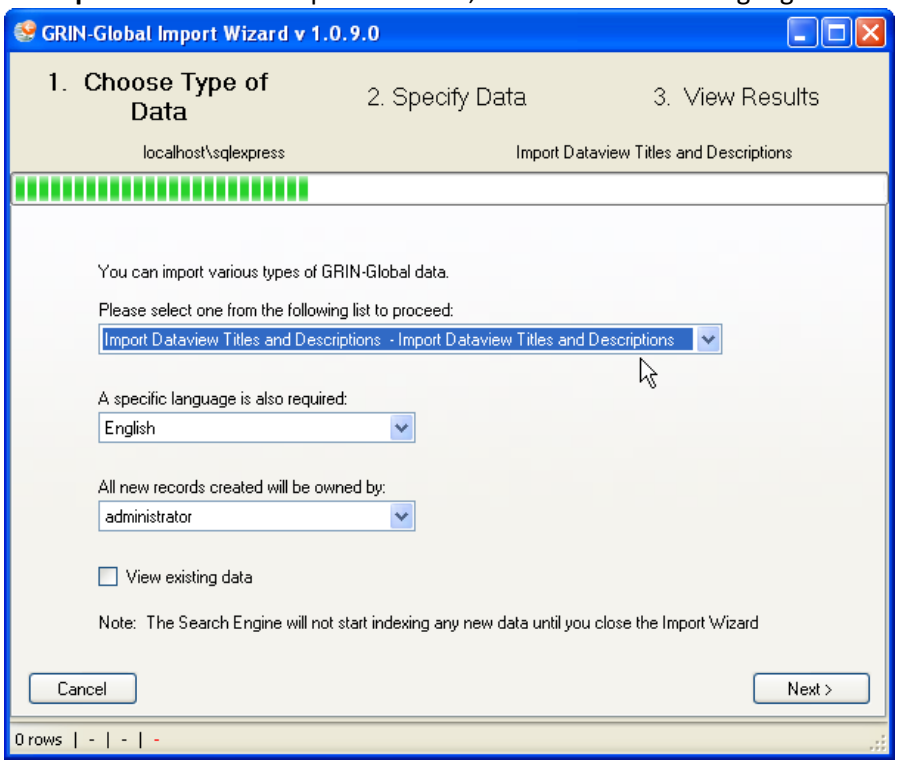
The resulting display will list the **Titles** and **Descriptions** as well as their English equivalents:



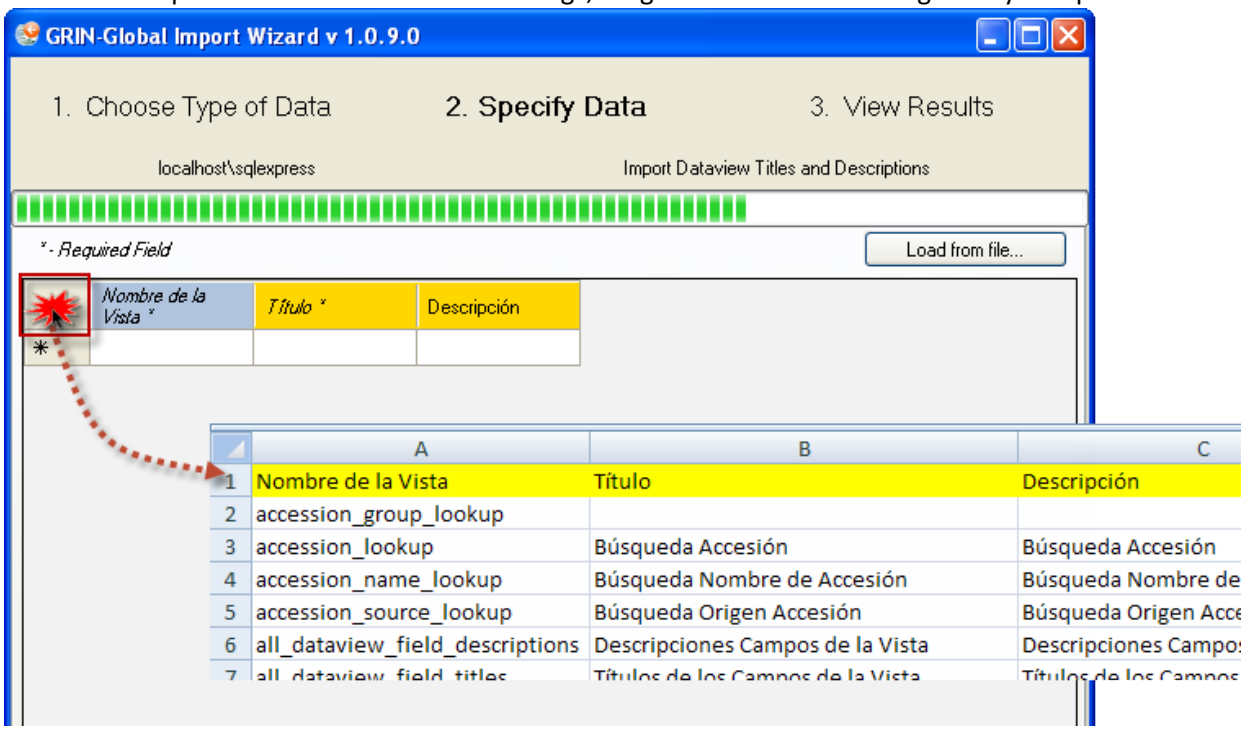
Copy these entries into a spreadsheet for review and editing. After editing, the next step is to import the data back into GG.

Importing Titles and Descriptions

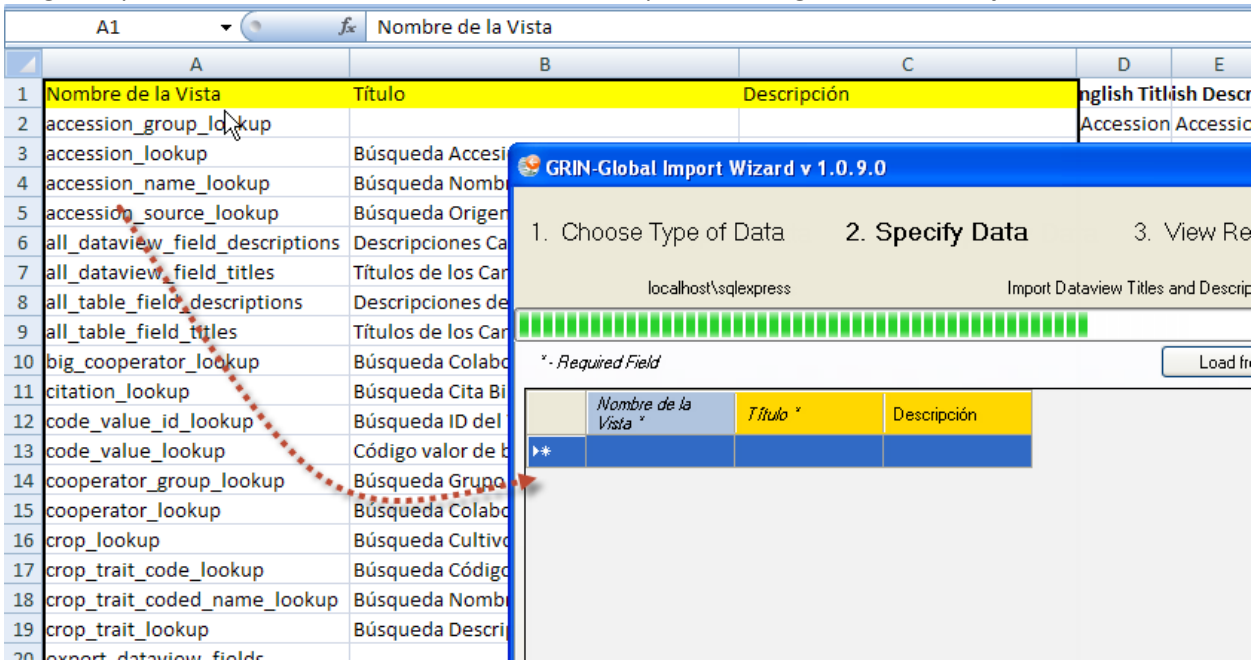
Use the Import Wizard **import Dataview Titles and Descriptions** dataview to import the Titles and Descriptions from the spreadsheet. Start the Import Wizard; select **Import Dataview Titles and Descriptions** from the dropdown menu; select the desired language:



Click on the top left corner to select the headings; drag the translated headings into your spreadsheet:



Drag the spreadsheet's first three columns into the Import Wizard grid; click the **Import** button:

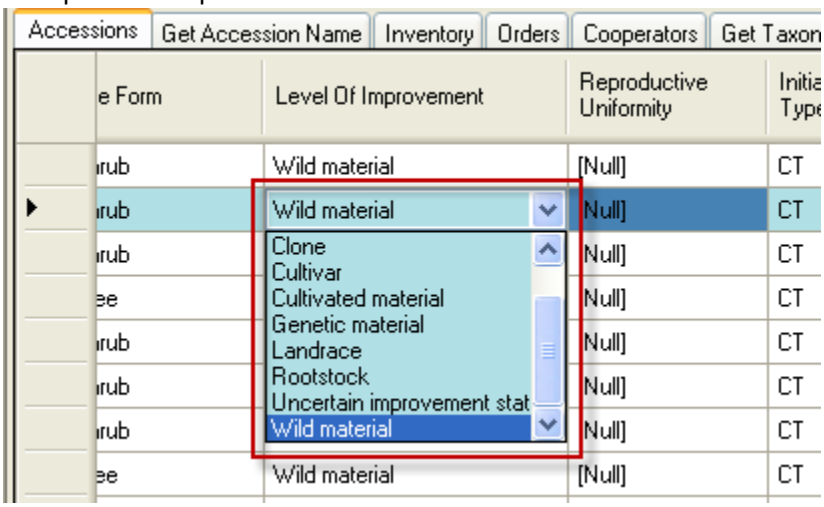


Code Groups

Overview

Many of the dataviews in the Curator Tool use dropdowns to assist the user in selecting a valid entry – the fields using dropdowns do not allow any random text data to be entered, but instead require a value from a pre-populated set of values. These values can be set up for each language that is installed in GRIN-Global.

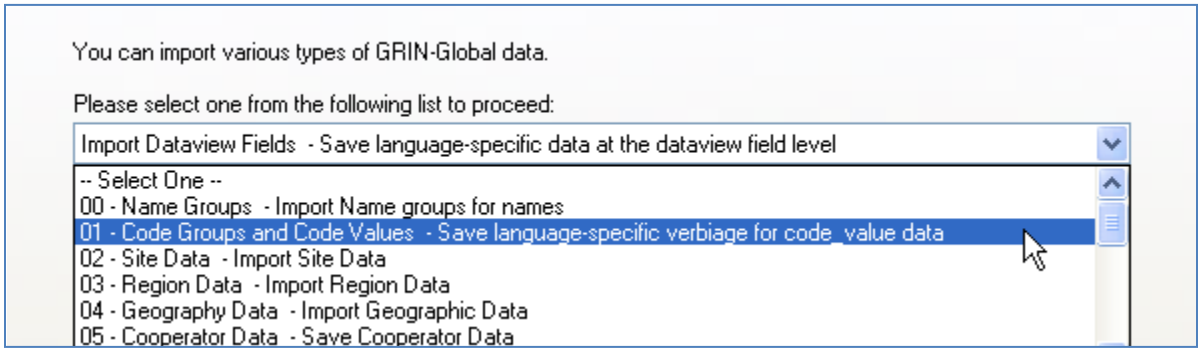
Example of a dropdown based on Codes:



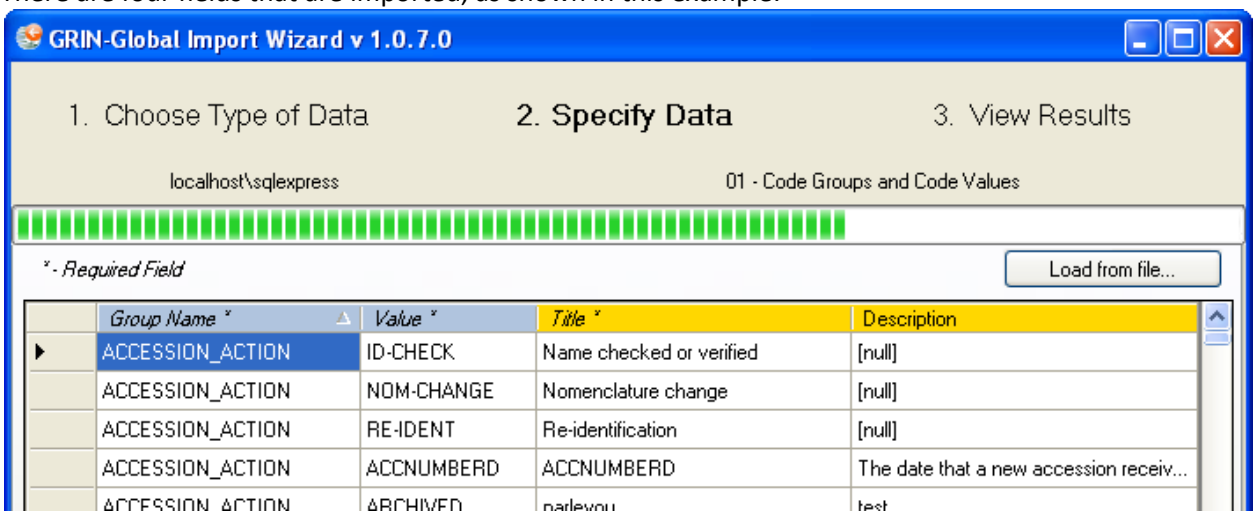
In the *Database Code Groups & Values GRIN-Global* section of the **Setup Guide Cookbook**, there is a detailed example illustrating how one Code Group, "IMPROVEMENT_LEVEL," is set up in the Admin Tool and then used in the Curator Tool.

Loading en Masse

Directions are also included there for initially bulk-loading **Code Groups and Code Values**, using the Import Wizard. Basically, groups of codes can be copied and pasted via the Import Wizard (IW). This IW dataview displays in the dropdown:



There are four fields that are imported, as shown in this example:



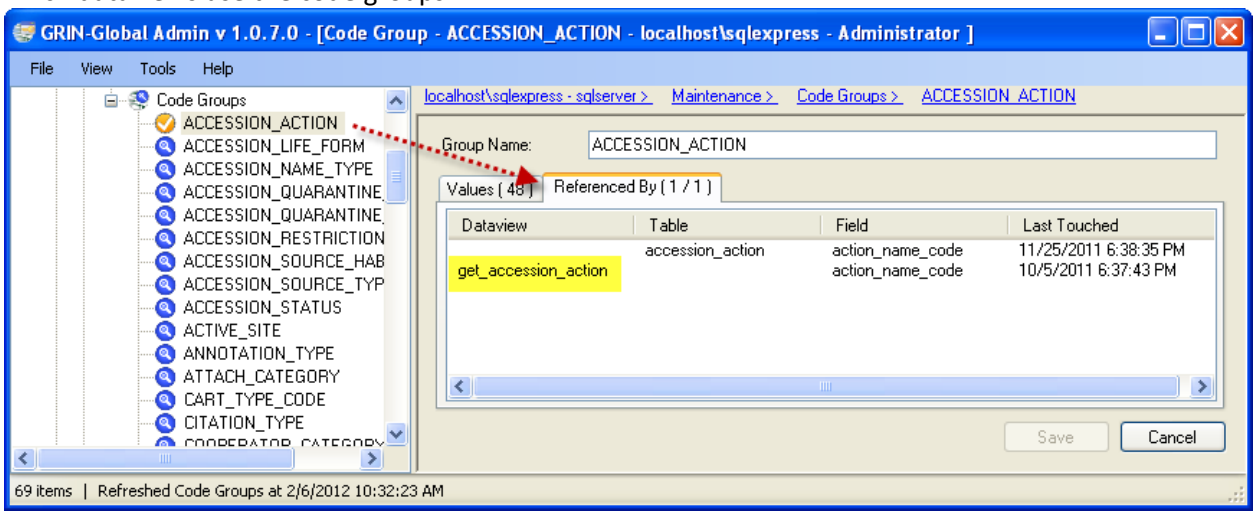
Editing Codes Individually

In the **Admin Guide**, there is a section titled *Code Groups*. There you will find detailed instructions for editing an existing code value. The section also explains how to quickly create a spreadsheet of code values for any Code Group, using the Admin Tool Code Group editor. This list can be modified, and then imported back into GG via the Import Wizard.

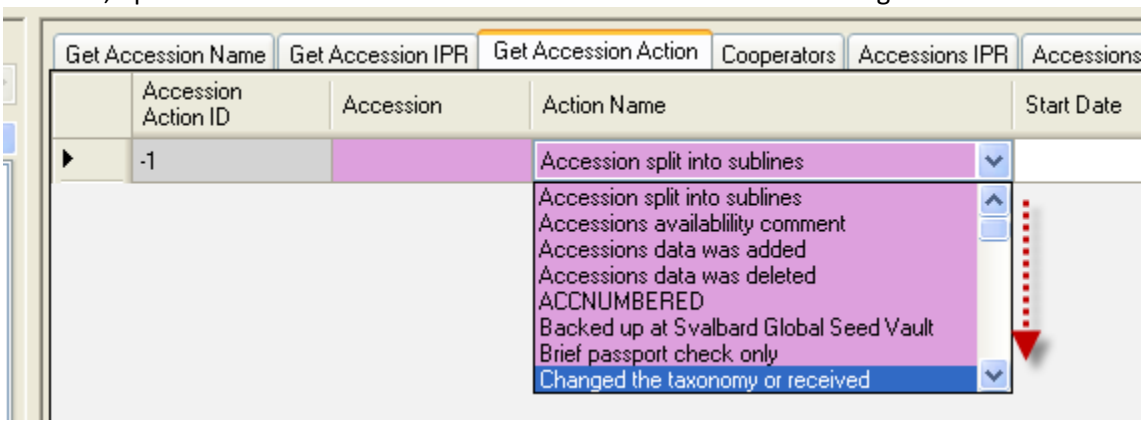
The Import Wizard doesn't delete records, so any codes that need to be deleted must be deleted via the Admin Tool editor.

Determining Where Code Groups are Used

Use the Code Groups tool in the Admin Tool to review the values for each group and also to review which dataviews use the code groups.



In the CT, open a dataview in Edit mode to review how the codes are being used:



Appendix

Deleting a Dataview

Overview

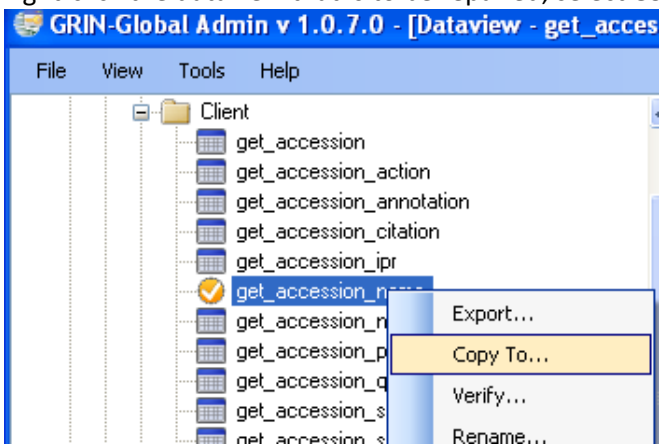
Whenever a dataview has a friendly name added or edited, a record is added to the **dbo.sys_dataview_field_lang** table in the database. If for some reason the intent is to have that friendly name removed from this table, it will be necessary to delete the modified dataview and then import the dataview back into GG.

Step-by-step Directions

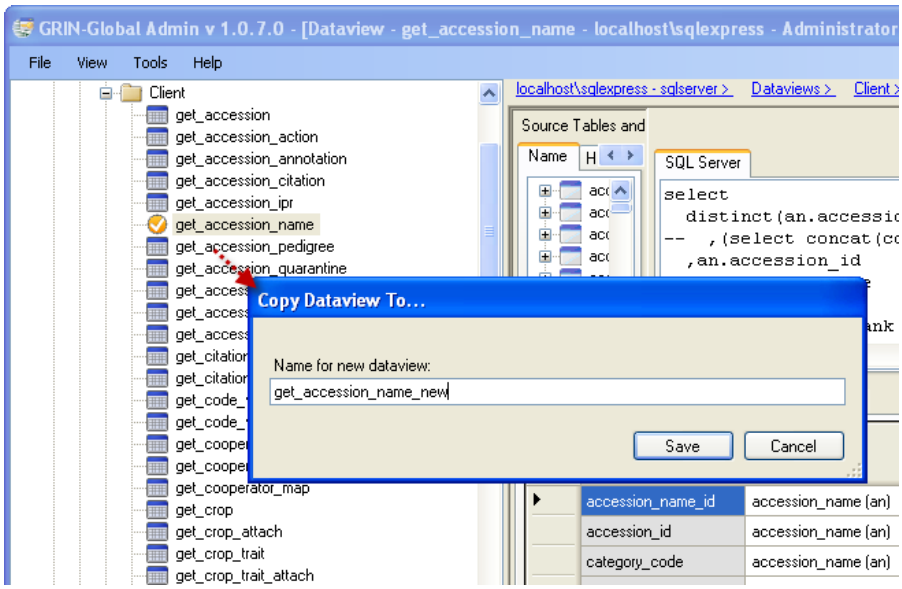
These directions guide you through making a copy of a dataview before editing the original one. This ensures that the original can be restored if needed for any reason. In the process, you will have two similar dataviews, the original and a copy.

1) In the AT, select the dataview that is to be corrected. (For example, a dataview with redundant language-friendly column names.)

2) Right-click the dataview that is to be repaired; select **Copy To**:



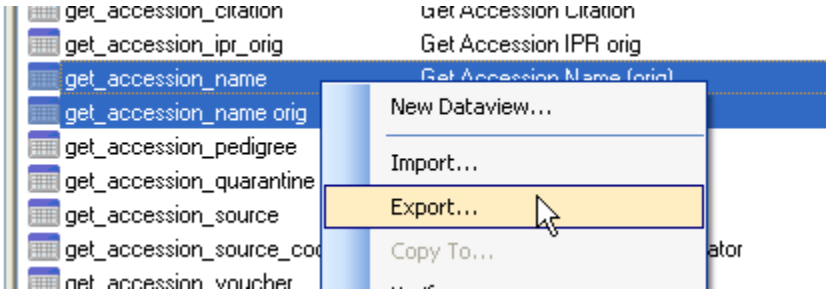
and provide a temporary name for this copy:



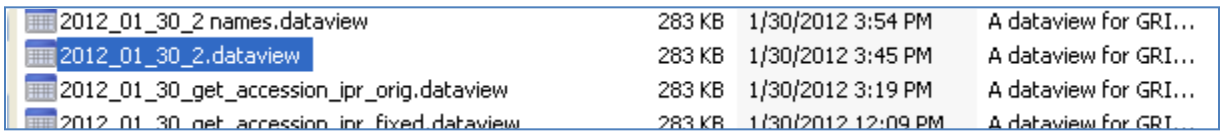
3) Select the new (copied) dataview. Make your corrections to the copy.

In any dataview, each language-friendly column heading name must be unique.

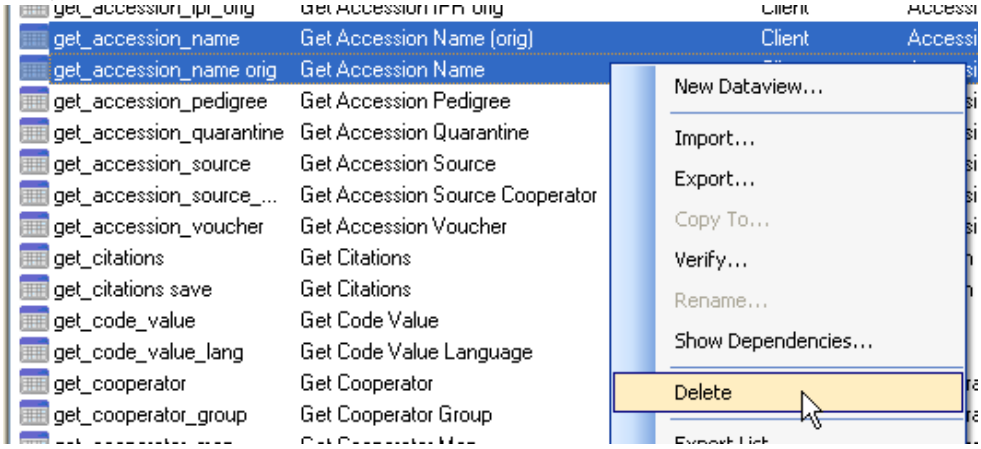
4) Keep both dataviews, the original and the copy, until you are certain everything is working as you expect it to. To do that, in the list view (the right grid,) select both dataview names, right-click, and select **Export...**



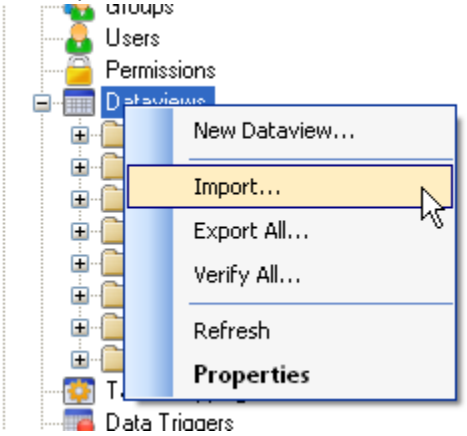
The dataviews will be saved on a drive with the extension "**dataview**." When you save a dataview for the first time, specify a dedicated folder that can serve as the holding place for any GG dataviews saved in the future. (The folder can hold other items, but it will be easier to view the dataviews if that is the sole purpose of the folder.) After the first save, GG by default will display that folder again as the potential folder whenever you attempt to save a dataview file.



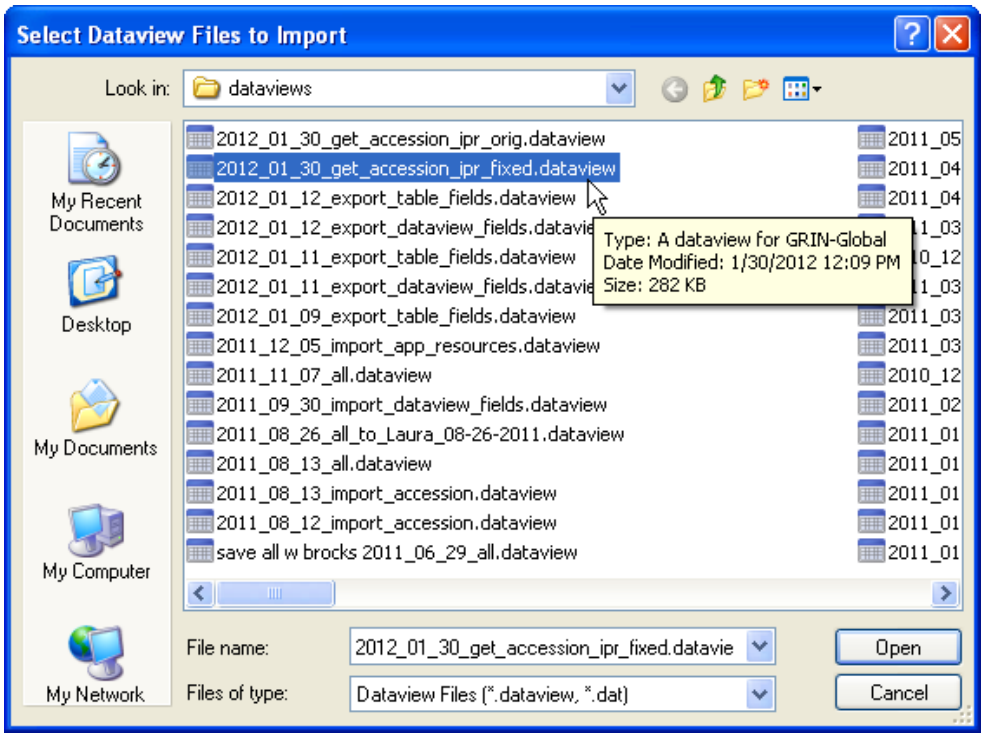
5) Delete *both* dataviews. To do that, in the list view (the right grid), select both dataview names, right-click, and select **Delete**



6) Now import the corrected dataview. At the **Dataviews** level, select **Import...**



7) Select the dataview with the corrections:



8) In the AT dataview editor, rename the corrected version with the original dataview name.

When you are satisfied with the new dataview, there is no need to keep storing the original dataview file. You can delete it from your drive or just hold onto it for awhile as a backup of the original.