The Ultimate Keyboard Remapper for Windows 9x and Windows NT
Version 2.1

User's Guide

GDG
SYSTEMES

GDG Systèmes, inc.
110, McGill Street   suite 202
Montreal,   Quebec
H2Y 2E5
Canada
Tel. (514) 875-0973
Fax. (514) 875-0974

KeyGO - Table of Contents

Introduction

Thank you for purchasing KEYGO version 2.x for Windows 9x and Windows NT. The goal of GDG is to provide you with a high quality keyboard remapper tool which is both powerful and easy to use. We share your commitment to quality and excellence, and will do our best to continue to provide you with software which meets your needs. We welcome any comments or suggestions.

KeyGO Overview.

KeyGO, a "Keyboard Redefinition Facility" is the successor of KeyGO version 1.x for Windows 3.1x and KRF Version 1.1. It allows you to remap keys on your keyboard to any character from the Windows ANSI character set.

Most often, KeyGO operates in the background of Windows. It translates your keystrokes into characters, strings, macros or OLE objects used in other Windows applications.. You can also launch an application, or a document, with keystrokes.

The KeyGO Builder allows you to construct your own keyboard. Every assignment is done using a Drag-and-drop process from any source that supports this process (the KeyGO Character map and Custom string box, Windows wall-paper, Windows Explorer or any Windows applications that supports OLE). KeyGO also lets you define an unlimited number of deadkey levels, making it one of the best utilities to use with phonetic or multiple glyphs languages on a US-type keyboard.

The Staff:

Programmers:	Steven Nathan
	Norbert Desautels
Manuals and Help files:	Norbert Desautels
	Wade Eide
	Corinne Gaudin
	Thierry Desautels
	Simon Allard
	Anne-Marie Trepannier
Logo and icon design:	Peter Paul Fianu

Getting Started

You are probably anxious to learn about KeyGO and try it out. We recommend that you follow these steps:

1. Fill out your registration card (before it gets lost!) and mail it to us.

2. Install Microsoft Windows 9x or NT (purchased separately) and learn how to use it. Do not attempt to install or use KeyGO until you are comfortable using Microsoft Windows.

3. Install KeyGO on your hard disk (See installing KeyGO page 3).

If you have any questions about a feature of KeyGO, please consult this manual first. If you can't find the answer to your question here, give our technical support a call, a fax or even better an e-mail. We'll try our best to help you.

> GDG Systems, inc.
> 110, McGill Street, Suite 202
> Montreal, Quebec
> CANADA   H2Y 2E5
> Tel.: +1 (514) 875-0973
> Fax.: +1 (514) 875-0974
> e-mail: support@gdgsystems.com
> Web: http://www.gdgsystems.com

Installing KeyGO

System Requirements

To use KeyGO, you need at a minimum:

- A 386,- 486-or Pentium based CPU.

- Windows 9x or Windows NT. You <u>cannot</u> run KeyGO 2.x under Windows 3.0 or Windows 3.1x.

- At least 4 MB of memory. This memory is required to use the Windows TrueType fonts.

- A graphics monitor, either color or monochrome, supported by Microsoft Windows 9x with at least VGA resolution. A color monitor is recommended for the keyboard builder.

- A pointing device supported by Microsoft Windows.

Note on Windows

Microsoft Windows must be installed and working before installing KeyGO. Windows 9x or NT is mandatory for KeyGO 2.x to run properly.

Installing KeyGO

Before beginning the installation procedure, please take a moment to read this entire section and the README.TXT file (if it exists) on the distribution disk. It is an ASCII file, so use any editor, such as Windows Wordpad, to view it. This file will contain last minute up-dates which you may want to know before installing the program.

KeyGO requires 2 Megabytes of free disk space.

Installing KeyGO from the diskettes

Run the installation directly from the Windows Task Bar. To begin, select **Run** from the **Start button** and type the following in the *command line* field of the dialog box:

A:SETUP

Click on **OK.**

Substitute the appropriate floppy drive letter if it is something other than the letter "A".

Another way to install keyGO is to choose the *Add/Remove programs* from Windows' *Control Panel* and click the **Install** button. Run the Setup program from the appropriate drive.

Once the installation procedure begins, an initial screen appears along with a timer bar. When it begins, read the options carefully and select the ones you want. You can specify drives and directories for KeyGO files. Once the installation is complete, a single new program group will be created, represented by the KeyGO icon.

Installing KeyGO from the self-extracting executable

Execute the file KEYGO21.EXE from the directory where it was downloaded.

Once the installation procedure begins, an initial screen appears along with a timer bar. When it begins, read the options carefully and select the ones you want. You can specify drives and directories for KeyGO files. Once the installation is complete, a single new program group will be created, represented by the KeyGO icon.

Windows NT 3.51

For Windows NT 3.51 users, the setup program can be run by choosing the **Run** option from the **File** menu in the **Program Manager** and typing **A:SETUP** Enter (replace A by B if your installation diskette is in drive B).

Language options and some View preferences will not be available under Windows NT 3.51, but only under Windows 9x, NT 4.0 and later.

The KEYGO.INI file and Windows registry

Starting with version 2.1 the KEYGO.INI file has been replaced by entries in the Windows registry. These entries will be automatically written by KeyGO. It contains several entries which you can customize (these entries are updated automatically by KeyGO and should not normally be changed manually). These affect certain aspects of the program's operation. Consult Appendix A (page 54) of this manual for detailed information.

Unlocking the release version of KeyGO.

KeyGO will run in demo mode the first it is launch. In this mode you won't be able to generate native keyboard drivers and a dialog box will pop-up every 50 typed character, reminding you of the demo status of KeyGO. To unlock the release version, click on the file menu and choose *Unlock KeyGO*. Enter the key code that was provided to you by GDG Systems or your distributor and click OK.

Keyboard file samples

Your distribution disks come with a set of keyboard template samples. These keyboards are the KeyGO equivalents of Windows' standard keyboard layouts. They can be used as a canvas on which to build your personnal assignments.

Note that those keyboard templates do not have any language assigned to the template, or script fonts attached to the keys. If you intend to use one or more of these keyboards as canvasses, choose a language in the **Language** pull-down menu in the **Tools** menu before using them (see page 46). You should also choose the font language from which you want to assign characters by using the **New font for all character assignments** from the **Tools** pull-down menu (see page 46).

You will also find in the set of samples a keyboard file named **FullAssgn.kgo**. This keyboard file gives you an example of a fully configured French-Canadian keyboard template (with function keys, deadkeys, toggle keys and character assignments).

Compatibility with old KeyGO versions

KeyGO version 2.x is not fully compatible with every version of KeyGO's file formats. If you have any problem importing your files, you should remove all the strings from within the previous version before opening them with KeyGO version 2.x.

Terminology

You should be familiar with the following terminology used to describe mouse and cursor actions:

•	*Primary mouse button*:	This is the mouse button you access with your index finger. It is often referred to as the left mouse button. However, this assumes right-handed mouse operations. Windows allows you to swap mouse button operations symmetrically, as an aid to left-handed people. To make the present manual applicable to both modes of operation, we refer to the button pressed by the index finger as the *primary* mouse button, regardless of the orientation.
•	*Secondary mouse button*:	This is the mouse button you access with your middle finger. It is often referred to as the right mouse button or the context menu button.
•	*Click On*:	Position your cursor on top of the designated item, then press and release the primary mouse button.
•	*Double-Click On:*	Position your cursor on top of the designated item, then press and release the primary mouse button twice in quick succession. If the two clicks occur within a specified time, Windows interprets it as a single action, referred to as a "double-click ". The *Mouse* option in the C*ontrol Panel* program, accessed using the *My Computer* Shortcut, allows you to lengthen or shorten the specified time period, according to your preference.
•	*Drag and drop:*	Depressing the primary mouse button, and while continuing to hold the button down, moving the mouse. When the desired action is completed, the mouse button is released. *Drag and drop* refers to the complete action: pressing and primary mouse button down, moving the mouse and releasing the mouse button.

For a more detailed discussion of Windows terminology, consult your " Microsoft Windows User's Guide ".

Uninstalling KeyGO

To uninstall KeyGO, choose the *Add/Remove programs* from Windows *Control Panel* and choose KeyGO from the list of programs. Then click the **Remove program** button. Or, for Windows NT 3.51; choose the Uninstall program in the KeyGO window group.

Starting KeyGO

To start KeyGO:

- After Windows has been started, and if the option *load automatically at Windows Startup* was validated in the Startup Options from the Options pull-down menu, KeyGO will be loaded automatically by Windows upon start-up.

- Otherwise, double-click on the KeyGO icon in the KeyGO program group.

To start the KeyGO keyboard builder:

- The first time KeyGO is run, the builder will come up automatically.

If KeyGO is already loaded, click on the KeyGO icon at the bottom of your screen or click the access icon in the System Tray of the Taskbar (for Windows NT 3.51 press Ctrl-Esc to get to the Windows *Task List* manager).

Double click here to launch KeyGO

KeyGO help file

Last minute information on KeyGO

Screen Layout

The KeyGO builder screen consists of the following key components:

- Windows border
- Title bar
- Menu bar
- Toolbar
- Status bar
- Horizontal/Vertical Scroll bar
- Editing Windows (layout background)
- Character map (optional)
- Customs string table (optional)
- Floating keyboard (optional)
- Floating Action bar (optional)

A description of each component follows:

Toolba   *Menu Bar*   *Title Bar*

Keyboard Layout   *Custom String Box*

Characters Box   *Deadkey list box*

Status Bar   *Action Bar*
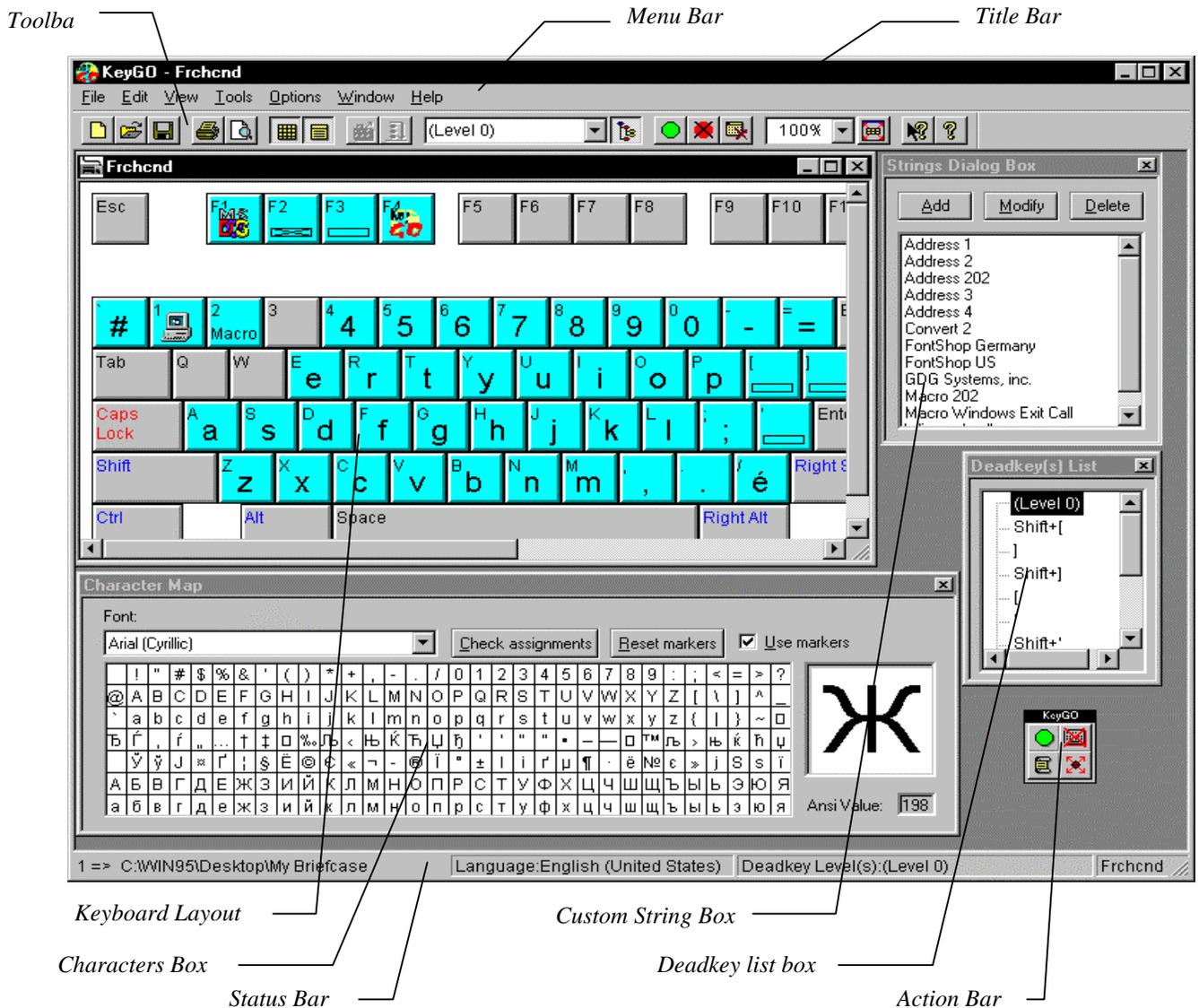
Windows Border

The Windows border is used for scaling the size of the KeyGO window. This is useful if you have other Windows applications running. To be able to use this feature, KeyGO must be in the *restore* window mode (press the icon on the title bar to shift to this mode). To re-size the window, use the size option in the system control menu or drag and drop the window border using the primary mouse button.

Title bar

In addition to telling you the name of the program and the name of the keyboard layout file you are working on, the title bar can be used to move the position of the KeyGO window on the screen.

The arrow icons on the right end of the title bar allow you to convert the KeyGO window to full size or reduce it to an icon.

The title bar contains the following elements:

- Application Control-menu button

- Document Control-menu button

- Name of the application

- Name of the keyboard layout

- Maximize button

- Minimize button

- Restore button

Toolbar

The Tool bar gives you quick access to the most common operations in KeyGO. The Toolbar contains the following tools:

	Create a new keyboard layout file (see page 37).
	Open a existing keyboard layout file (see page 37).
	Save the current keyboard layout file (see page 38).
	Print the keyboard layout assignment list (see page 38).
	Print Preview of the keyboard layout assignment list (see page 40).
	Hide or show the character map (see page 43).
	Hide or show the Custom string table (see page 43).

	Unassign Key (see page 45).
	Make deadkey (see page 40).
[None]	Pull-down menu with a list of the current available deadkeys (see pages 22 and 26)
	Show or Hide the Deadkey list box from which the deadkey level can be selected (see page 25).
	Activate current keyboard layout (see page 44).
	Deactivate KeyGO (see page 44).
	Hide or Show the floating keyboard (see page 43)
150%	Pull-down selection menu of the display size for the current keyboard layout (see " Scale to fit " page 43).
	Scale to fit the current keyboard layout (see page 43).
	Context-sensitive help (see page 16).
	Bring up the *About KeyGO* Box

You can also get fast help on each of the different toolbar icons by placing the mouse arrow on the icon and waiting for two (2) seconds for the *Tooltips* box to pop up with the description of the command.

The toolbar can also be moved to a floating position, or to another border location, by dragging and dropping its frame. It can also be resized by dragging and dropping its border.

Menu bar

The menu bar contains the following drop-down menus:

File

New	Ctrl+N
Open...	Ctrl+O
Close	
Save	Ctrl+S
Save As...	
Print...	Ctrl+P
Print Preview	
Print Setup...	
1	
2	*Recent Files*
3	
4	
Exit	

Edit

Cut	Ctrl+X, Del
Copy	Ctrl+C
Paste	Ctrl+V
Insert New Object...	
New File assignment...	
Launch Explorer...	

View

✔ Toolbar	
✔ Status Bar	
✔ Character Map	
Strings Box	
Deadkey List	
Floating Keyboard	
✔ Action Bar	
Scale to Fit	
Zoom	▶

Tools

Activate [Frchcnd]	Ctrl+K
Deactivate KeyGO	
Unassign	
Deselect All	
Make Deadkey(s)	
Make Toggle Key(s)	
Copy Assignments to New Keyboard	
New font for all character assignments	
Language	▶

Options

Startup Options...
Keyboard Mapping...
Keyboard Summary...
Keyboard General Options...
Keyboard Language Options...
View Preference...
OLE Transfer Options...
Character Transfer Options...

Window

New Window
Cascade
Tile
Arrange Icons
✔ 1 Frchcnd

Help

Help Main Index
Help Topics
Using Help
About KeyGO...

Detailed descriptions of each of the commands found in these menus are giving in the Menu Commands chapter (page 31).

Horizontal/Vertical Scroll bars

Scroll Bars are provided to pan your current viewing window. Panning is particularly useful when you are using multiple keyboard layouts. The scroll boxes inside the scroll bars indicate your vertical and horizontal location on the keyboard layout. You can use the mouse to scroll to other parts of the keyboard layout. (See also "Scale to fit", page 43.)

Editing Window

The large white portion of the screen is the editing window. The keyboard in the center represents your working area. The Editing Window can be used to eliminate characters that no longer need to be assigned to a key by dragging and dropping the assignment from the key to the Editing Window background.

Status bar

```
F4 => C:\KEYGO\KEYGO.EXE          Language:English (United States)  Deadkey Level(s):(Level 0)          Frchcnd     CAP NUM
```
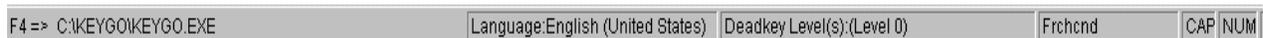
The status bar is displayed at the bottom of the KeyGO window. To display or hide the status bar, use the Status Bar command in the View menu.

The left area of the status bar describes actions of menu items as you use the arrow keys to navigate through menus. This area also shows messages that describe the actions of toolbar buttons and menu options as you depress them. If, after viewing the description, you do not wish to execute the command, move the pointer off the toolbar button or the menu option and release the mouse button. This area also shows the assignment of the key under the mouse cursor.

The middle part of the Status Bar indicatesthe language that defines the current keyboard layout, the current deadkey level, and the name of the current active keyboard.

The right-hand area of the status bar indicates which of the following keys are latched down:

Indicator	Description
CAP	The Caps Lock key is latched down.
NUM	The Num. Lock key is latched down.
SCRL	The Scroll Lock key is latched down.

Character map

The character map is one of the possible sources from which the key assignments are made. It is used to make simple character assignments.
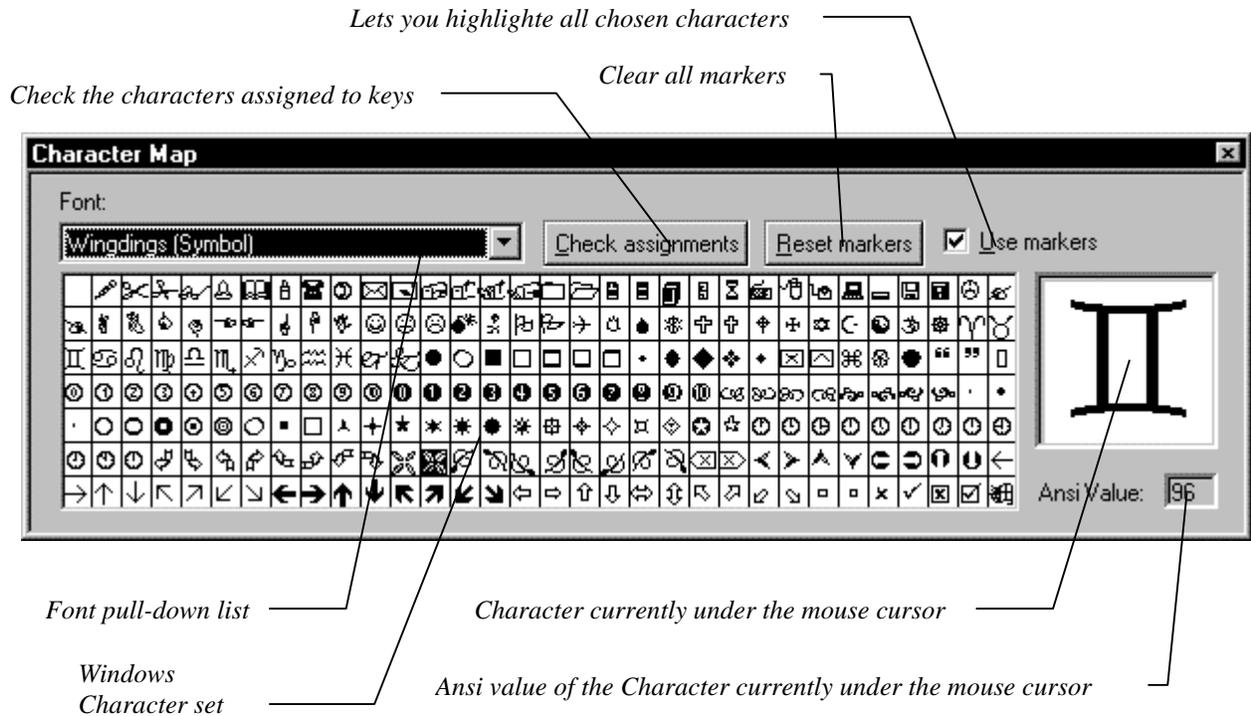
Lets you highlighte all chosen characters

Clear all markers

Check the characters assigned to keys

Font pull-down list

Character currently under the mouse cursor

Windows Character set

Ansi value of the Character currently under the mouse cursor

To activate the character map, click on the Character Map Icon on the toolbar [icon] or select **Character Map** from the **View** menu. Use the same options to close the character map, or double-click on the Close icon in the upper right corner of the character map table.

In normal use, you will first select the font from the pull-down menu in the character map, then click-hold on a character from the same character map, drag the mouse to a key on a keyboard layout, and drop the character.

The **Font** pull-down box lets you choose the font that you need for the assignments.

The **Check assignments** button when clicked will underline in red all the characters that are currently assigned to a key (or any level of state key or deadkey). These checks will not be updated automatically with each new assignment. To have a automatic update of assignments you should use the Check assignments feature, combined with the **Use marker** option (see below).

The **Reset markers** button will clear all underlining and markers from the current font character set.

When the **Use markers** is checked, characters (from the character map) when clicked will be highlighted (show in reverse video). To remove a marker from a character, you can either click it again, or use the **Reset markers** button.

Custom String

The Custom String Table is one of the possible sources from which the key assignments can be made. KeyGO lets you assign strings of characters that can contain special controls making it possible to build macros for all Windows applications (see String Assignment page 18).

Using the KeyGO On-line Help

KeyGO comes with an extensive on-line Help file which provides instructions on how to perform each program operation. You can access the on-line Help file in a number of ways. The most rudimentary method is by pressing F1 while no dialog box is open or menu command highlighted.

KeyGO also provides context-sensitive Help. You can access it in two ways. The first is available only when a dialog box is open, or when an item under one of the main menu commands is highlighted. Then press F1. The second method is used when no dialog boxes are open and no menu items are highlighted. In this case, hold down the SHIFT key and press F1 or click on the [icon] icon on the Tool bar. The cursor changes to a [cursor] . This special cursor allows you to access help on a single item by positioning it over the item on the screen and clicking.

You can also get fast help on each of the different toolbar icons by placing the mouse arrow on the icon and waiting for two (2) seconds for the *tooltips* box to pop up with the description of the command.

Exiting KeyGO

Selecting **Exit** from the **File** menu ends the current KeyGO builder session and unhooks KeyGO from the Windows keyboard. To exit the KeyGO builder while keeping the KeyGO translator active, click on the minimize button [button] located on the Title bar, or use the proper button on the Action Bar or the Windows 9x System Tray. If you Exit KeyGO while it is still active this pop up window will appear:

From there you can choose to have KeyGO turned off by choosing Quit KeyGO or click **Background** to keep it active in the background. The Cancel button lets you go back to the builder.

Using KeyGO (Quick Start)

KeyGO, a "**K**eyboard **R**edefinition **F**acility" is the successor of KeyGO version 1.x and KRF. It allows you to remap keys on your keyboard to any character from the Windows ANSI character set or to assign strings with special control codes.

At runtime

Most often, KeyGO operates in the background of Windows. It translates your keystrokes into any KeyGO assigned object (characters, strings, macros, file launching and OLE object used in other Windows applications).

Once a keyboard file has been constructed (see the chapter "Using the KeyGO builder") and saved, you can make it your default keyboard layout by following these steps:

1. From the **Options, Startup Options** submenu (see page 46), click the **Browse** Option and choose the layout file that you want to use under Windows

2. Click the **Start in background mode** box and **the Load KeyGO automatically at Windows startup** box..

3. Click - **OK**.

4. Click on the minimize button ▬ located on the Title bar if you want to put KeyGO in the background for this Windows session.

From now on, each time it is started, the KeyGO program will always be in the background with your default keyboard loaded.

Using the KeyGO Builder

The first time KeyGO is executed after the set-up process or after a double-click on the KeyGO icon, the KeyGO builder will come up on your screen. From there you can either use the File **Open** submenu to load one of the KeyGO standard keyboard layout files, or construct your own keyboard layout.

Using the mouse

Every assignment in KeyGO is performed using a Drag-and-Drop process from the Character Map, the Custom String box, all applications that can provide a file name (Ex.: Windows Shortcuts or Explorer) or any application that supports OLE 2.0

Placing the cursor arrow over the key on the keyboard layout and clicking on the primary mouse button does selection of a key. The purpose of selecting a key is to make a deadkey or to unassign characters; no selection of the key is necessary to make an assignment.

It is also possible to unassign a key by dragging-and-dropping the character from the key to the keyboard layout background.

A group of keys can be selected by dragging-and-dropping a virtual box, starting from the background of the keyboard layout (the white area), around the keys you want to select with the right mouse button pressed.

A **context menu** will be display when clicking the right mouse button over a key.

Builder Overview

The KeyGO Builder is a window with seven (7) main areas:

1. The application menu with **File**, **Edit**, **View**, **Tools, Options**, **Window**, and **Help** menu.

2. The Toolbar ribbon allowing you to select deadkeys and perform frequently needed functions.

3. The Keyboard Layout area which shows your keyboards and their assigned keys.

4. The Character Map from which you chose the characters you want to assign.

5. The Custom String box which allows the user to define characters and function strings.

6. The Status bar at the bottom of the KeyGO window which provides brief help messages, the status of some important keys and the name of the active keyboard layout file.

7. The deadkey(s) list box used to select the proper deadkey level. This list box is useful when more then 3 levels of deadkeys is necessary.

The first step in creating a simple character assignment KeyGO file is to select the font you want to use. This is done by using the font selection pull-down menu located on the Character Map.

From there, the assignment process is done quite simply by dragging the character you want to assign from the character map (see page 12) to the desired key on the Keyboard Layout (see page 17).

When a character is assigned to a key, you will note that the key is light blue. When a key has an assignment AND it is the currently selected key, it is dark red.

IMPORTANT: The SHIFT, CTRL and ALT keys are regarded by KeyGO as STATE keys. That is, they change the state of any other key that is selected. You can use any combination of these STATE keys. For example, if you want to make more assignments then the maximum number of keys on the keyboard, you can use the ALT, CTRL and SHIFT keys in any combination to create more assignment space.

If you wish to make a Deadkey (see page 22) select a key and press the **Make Deadkey** button on the toolbar or from the **Make Deadkey** option in the **Tools** menu. (**Caution:** make sure you properly understand every aspect of deadkeys before using them. See page 22).

You can also assign a String of Characters to a key (see page 18). This option is only available if the **String Box** option is selected from the **View** Menu. When the different strings have been constructed, the same process of drag-and-drop is used to assign these strings to their respective keys. All function and STATE keys can also be included in the assigned strings.

Continue this process for all keys you wish to assign.

When you are finished, be sure to save your file by picking the **Save** option from the **File** Menu. A description (Title, Keywords and Comments) for this KeyGO file can be set by picking **Keyboard Summary..** options from the **Options** Menu.

IMPORTANT: When using non-roman font types during the assignment process: Do not forget to choose the font in the Windows application used with KeyGO, corresponding to the one used in making the assignments. For example, if a special keyboard was made in KeyGO using the Wingdings font, the Windows application using this special keyboard would have to be set using the Wingdings font. Use the Auto activation string option in the **Keyboard General Options** dialog box to automatically switch your application to the desired font when using a different keyboard layout (see page 49). **However, for fonts with script definitions, KeyGo will automatically switch the application to the proper language/font (see Multiple language support page 31).**

Context Help command

Use the Context Help command to obtain help on any feature of KeyGO. When you choose the Toolbar Context Help button, the mouse pointer will change to an arrow and question mark. Then click anywhere in the KeyGO window, such as another Toolbar button or on a menu choice. The Help topic will be shown for the item you clicked.

Shortcut

Keys: SHIFT+F1

Toolbar:

Dragging and Dropping

In KeyGO, every assignment is made using the Drag-and-Drop process. Simply select a character from the character map, a string name from the Custom String table, a file name from an Explorer-like application, or an object (text, picture, sound, etc.) from an OLE 2.0 compliant application, drag the mouse to a key on a keyboard layout by keeping the mouse button pressed. Drop the text, string, file or object by releasing the mouse button.

You can unassign from a key by dragging and dropping the character from the key to the keyboard template background.

You can move assignments from key to key just by dragging and dropping them. If there is already a character assigned to the destination key, the two characters will exchange positions.

It is also possible, using the drag and drop process, to transfer characters, strings and deadkeys from one keyboard layout to another.

Keyboard Layout
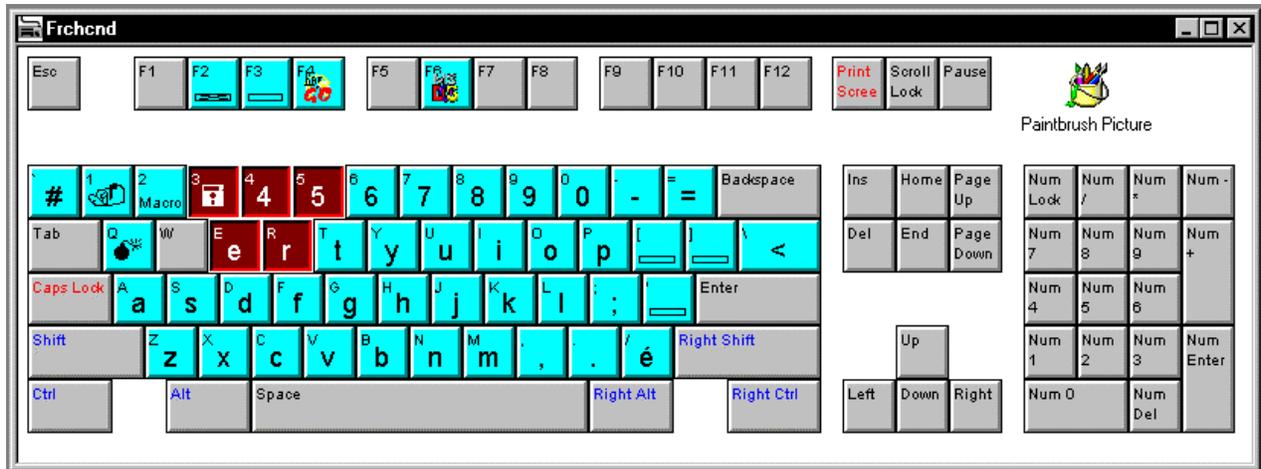
The keyboard layout is the target of the drag-and-drop process with which keys are redefined. It represents the physical keyboard actually in use. When creating a new layout, the Keyboard Type dialog box will pop up prompting for the type of physical keyboard currently used. If you are unsure of the type of keyboard, try a different type and compare the layout shown on the screen to the physical keyboard on your table.

Context menu

The right mouse when pressed over a keycap brings up a context menu that lets you change specific aspects of that key.
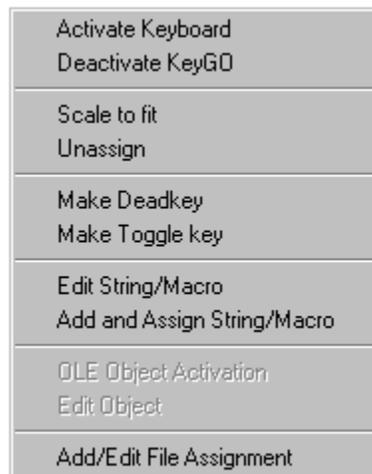
The **Activate Keyboard** and **Deactivate KeyGO** options will activate the current keyboard or deactivate KeyGO completely.

The **Scale to fit** option will resize the current keyboard layout to match the size of the surrounding window.

The **Unassign** option will remove any assignment present on the current key.

The **Make Deadkey** option will make the current key a deadkey (see Deadkeys Overview page 25).

The **Make Toggle** key option will make the current key a toggle key (see Making Toggle keys page 29).

The **Edit String/Macro** option will be available if the assignment on the current key is a string or a macro. This option lets you edit a string or macro without having to use the String dialog box.

The **Add and Assign String/Macro** lets you construct a new string and assign it to the currently selected key.

The **OLE Object Activation** lets you choose the way KeyGO will activate the OLE object present on the Keycap (see page 23).

The **Edit OLE Object** lets you modify the object from its original application.

The **Add/Edit File Assignment** lets you edit or assign a new file to the current key (see page 22).

You can also bring up a context menu in all KeyGO edit dialog boxes.   The menu options let you Undo your last action, Cut, Copy, Paste, Delete the selected text or Select the whole text.

Character Assignment

Every character assignment is made using a Drag-and-drop process from the Character Map to the keyboard layout(s).

To activate the character map, click on the Character Map Icon on the toolbar 🔲 or select **Character Map** from the **View** menu. Use the same options to close the character map, or click on the Close icon in the upper right corner of the character map table.

Do not forget that all SHIFT, CTRL, ALT and/or AltGr (see page 22) keyboard states can be used in any combination to assign a character. Clicking on one or more of these keys on the Keyboard Layout will toggle the state of the keyboard letting you make assignments on different levels.

First select the font from the pull-down menu in the character map, then click-hold on a character from the same character map, drag the mouse to a key on a keyboard layout, and drop the character.

Once all the assignments are completed, save your keyboard layout file using the **Save** or **Save As...** command from the **File** menu (see page 38).

Example: Let's assign the non-breaking space character to the shift-space bar key. First, click the Shift key on the keyboard template. Second, click the non-breaking space character from the Character map (fifth row, first column, ansi value: 160) and while holding the mouse button, drag the character to the space bar on the keyboard template. Now, the non-breaking space character can be use in all Windows applications by depressing the shift-space bar combination.

Important: To test or use your keyboard you must first activate it by using the 🟢 icon on the toolbar. To make your keyboard layout the default, to be loaded each time Windows is started, you must specify its name in the **Startup Options** dialog box from the **Options** menu (see page 46).

String Assignment

KeyGO lets you assign not only single characters to keys, but strings of characters as well. These strings can contain all the normal Windows ANSI characters, state keys (Ctrl, Shift and Alt) and special function keys ({Enter}, {Tab}, {F1} to {F16} , etc.).

NOTE: KeyGO version 2.x cannot send commands to the Operating System (ex.: Ctrl-Esc to call the Taskbar or Alt-Tab to switch applications) under Windows 9x and Windows NT.

To activate the custom string box, click on the Custom String Icon on the [icon] or select the **Strings Box** option from the **View** menu.

The scroll down window shows all the names of available strings. These strings can be assigned to a key on the keyboard by dragging and dropping the name of the string from the custom string table to the proper key on the keyboard. By using the **Add** button you can construct your own string.

The **Modify** button lets you edit the currently selected string. Do not forget that all SHIFT, CTRL and/or ALT keyboard states can be used in any combination to assign strings. Clicking on one or more of these keys on the Keyboard Layout will toggle the state of the keyboard, letting you make assignments on different levels.

Once all the assignments are completed, save your keyboard layout file using the **Save** or **Save As...** command from the **File** menu (see page 38). To test your keyboard you must first activate it by clicking on the [icon] icon on the toolbar, then use any Windows text editor.

Adding and Modifying strings

After clicking on the **Add** or **Modify** button from the custom string window, the edit string window comes up.

Note: In certain cases, it is best to deactivate KeyGO during string construction, since an active KeyGO will continue to affect your keyboard output.

Note: All the normal Cut and Paste Windows shortcut keys are available in these two dialog boxes (Ctrl-C = Copy, Ctrl-V= Paste, Ctrl-X= Cut). Multiple copies of a string can be made using this feature.
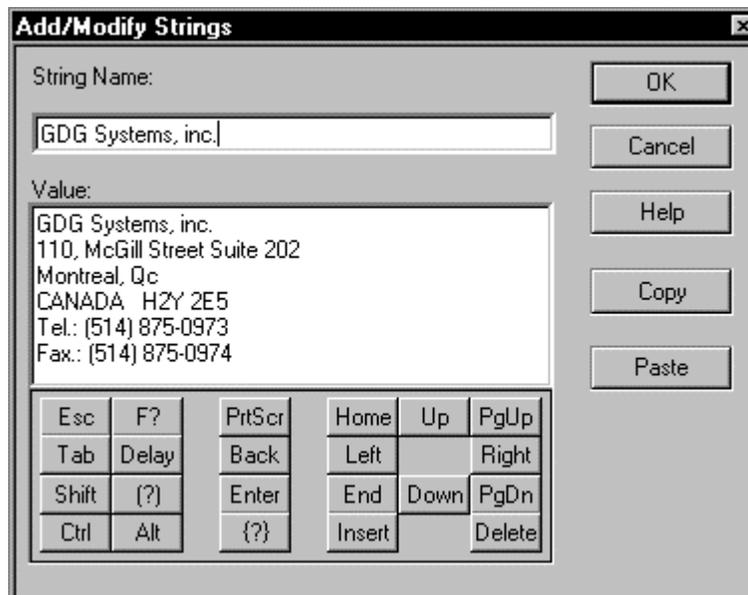
A context menu (right click on the content) that will bring up some editing options.can be used in the String Name and the Value box.

The **String Name** dialog box lets you give a name to the string as it will appear in the Custom String box.

The **Value** edit box is where the actual string will be entered or modified.

The plus sign (+), caret (^), percent sign (%), tilde (~), parentheses (), braces ({}) and brackets ([]) have special meanings to KeyGO. To specify one of these special characters as normal characters, enclose it inside braces. For example, to specify the plus sign, use {+}. To send brace characters, use "{{}" and "{}}", respectively.

Brackets ([ ]) are used to specify a delay in milliseconds after which the following character or function will be sent. This can be useful when making demonstrations that need special timing. Also, some Windows applications can not get all the messages sent to them at full speed (ex.: QuarkExpress). If you have such an application, placing a [1] (delay of 1 millisecond) in front of this character or function will normally do the trick. Increase this delay if the problem persists. To send brackets, use "{[}" and "{]}", respectively. The delay value can be between 1 to 65535 (1 minute 5 seconds).

Examples:[1000]Hello[2000]~ , The string "Hello" will be sent to the application after a delay of 1 second (=1000 Millisecond) followed by ENTER after a delay of 2 seconds.

It is also possible to combine brackets and parentheses (see below) to generate a string in which all characters and functions will be send to the application with the specified delay. For instance, all five characters of the string [1000](Hello) will be sent to the application with a delay of 1 second.

The lower part of the Add/Modify String window contains shortcut buttons for the special function characters. Clicking on these shortcuts sends the proper code to the Value dialog box without having to write them manually.

To specify Shift, Control and Alt keys, precede the regular key code with one or more of these codes:

Key   Code

Shift   +
Control   ^
Alt   %

To specify that Shift, Control, and/or Alt should be held down while several keys are pressed, enclose the keys in parentheses. For example, to hold down the Shift key while pressing E then C, use "+(EC)". To hold down Shift while pressing E, followed by C without the Shift key, use "+EC". To specify repeating keys, use the form {key number} where there is always a space between key and number. For example, {LEFT 42} means press the Left Arrow key 42 times; {x 10} means press the character x 10 times.

All normal characters can be used in strings. Special function characters may also be employed. Below is the list of the different special codes which can be used in the Value dialog box. Except for the state keys, %, + and ^, all should be entered enclosed in {}. If a bad string value is specified (wrong delay or control codes) it will be sent to the application without being interpreted.

Control Codes	Actions
%	The Alt state key. Use it to change the Alt status of the character or the bracket function that follows it.

^	The Ctrl state key. Use it to change the Ctrl status of the character or the bracket function that follows it.
+	The Shift state key. Use it to change the Shift status of the character or the bracket function that follows it.
~ (tilde)	Send the Enter command to the application (same as {ENTER}).
{ESCAPE}	Send the Escape character to the application.
{ESC}	Send the Escape character to the application (same as {ESCAPE})
{ENTER}	Send the Enter character to the application.
{PRTSC}	Send the Print Screen command to the application.
{TAB}	Send the TAB character to the application.
{BREAK}	Send the Ctrl-Break command to the application.
{BACKSPACE}	Send the Backspace character to the application (same as {BS} and {BKSP}).
{BS}	Send the Escape character to the application (same as {BACKSPACE} and {BKSP}).
{BKSP}	Send the Backspace character to the application (same as {BACKSPACE} and {BS})
{DELETE}	Send the Delete character to the application.
{DEL}	Send the Delete character to the application (same as {DELETE}).
{INSERT}	Send the Insert character to the application.
{LEFT}	Send the Left direction key character to the application.
{RIGHT}	Send the Right direction key character to the application.
{UP}	Send the Up direction key character to the application.
{DOWN}	Send the Down direction key character to the application.
{PGUP}	Send the Page Up key character to the application.
{PGDN}	Send the page Down key character to the application.
{HOME}	Send the Home key character to the application.
{END}	Send the End key character to the application.
{F1} to {F16}	Send the function key character F1 to F16 to the application

Table 1- String Control Code List

Deleting a string

To delete a string from the Custom string box, just click on it, then press the **Delete** button.

String examples

The following are a few examples of string values as entered in the String **Add/Modify** Value dialog box, along with their respective actions in the application.

Strings	Actions
Hello Word!~	- Send the phrase "Hello Word!" followed by the Enter key
{F1}	- Send the F1 key - Usually the Help key.
z{left}{del}ABCD	- Send the z character, move left, clear the z and write ABCD
{right 20}	- Move the cursor 20 times to the right
{{}Brackets{}}	- Send the {Brackets} string to the application
[5000]%{F4}	- Send the Close Application command (Alt-F4) to the application after a delay of 5 seconds.
%f	- Alt-F
%of%f%pSymbol{tab}{tab}12{enter}	- For Winword 6.0 - Shift to the Symbol font, 12 pitch size.

Table 2- Control Code Examples

File Assignment

The goal of making File assignments is to be able to launch applications or to fast load documents from a key or a sequence of keys (if used in conjunction with Deadkeys. See page 30). In essence these assignments work like the Shortcut key of Windows 9x.

File assignments can be made in three different ways.

NOTE: The first way (see below) of making file assignments is the only way to assign **multiple** files or directories to a key.

1. By using the drag-and-drop process from an Explorer-like application to the desired keycap on the keyboard template. You can also drag-and-drop files from their shortcut locations on the wall paper of Windows 9x. **Example:** Launch the Windows Explorer program from the Start button and place KeyGO and Explorer side by side on the screen. Now, choose from Explorer a file (or a group of files) that you want to assign to a key and drag-drop it on the desired key on the keyboard template of KeyGO. It is also possible with this method to assign multiple files to a key. To do so, select the files in the Explorer program using the normal keyboard Windows command to select multiple file (Shift or Ctrl + Click) before doing the drag-and-drop process. After the keyboard has been activated, depress the assigned key to launch the application(s) or to load of document(s). Using this method of assignment, you can also assign a directory tree. Depressing the key on which a directory tree is assigned will open a window containing the folders and files currently under the specified directory, letting you access them rapidly. **This is the only way to assign multiple files or directories to a key.**

2. By using the context menu. Right click on the desired keycap. A menu will appear. From this menu select the **Add/Edit File Assignment** option. A standard Open dialog box will appear, letting you choose the desired file.

3. By using the **New File Assignment** option from the **Edit** pull-down menu. The file that will be selected with this option will be assigned to the first vacant keycap on the keyboard layout. From there it can be moved to the desired keycap using the drag-and-drop process. Note that you should first select the proper Ctrl-Alt-Shift state and deadkey level of the keyboard before using this option.

Do not forget that all SHIFT, CTRL, ALT and/or AltGr (see page 22) keyboard states can be used in any combination to assign a character. Clicking on one or more of these keys on the Keyboard Layout will toggle the state of the keyboard, letting you make assignments on different levels.

Example: For programmers, who frequently use the same project file, or for a secretary, who often uses the same document template, assignment of the document to a key can save a great deal of time by launching the corresponding application (with the proper document ready for editing) each time the assigned key is depressed.

Once all the assignments are completed, save your keyboard layout file using the **Save** or **Save As...** command from the **File** menu (see page 38).

Important: To test or use your keyboard you must first activate it by using the 🟢 icon on the toolbar. To make your keyboard layout the default, to be loaded each time Windows is started, you must specify its name in the **Startup Options** dialog box from the **Options** menu (see page 46).

After the keyboard has been activated, depress the assigned key to launch the application or to load the document.

OLE (Object Linking and Embedding) 2.0 Assignment

With the Object Linking and Embedding (OLE) features in KeyGO, you can enhance your keyboard assignments to include parts of documents (with all attributs), logos, sound, etc. .

IMPORTANT: Once an object has been assigned to a key, you can double click on this keycap with the left mouse button (or using the context menu. See page 17) to change how KeyGO will process the OLE object or to edit the OLE object. You can also bring up the context menu (right clicking on the key. See page 17) and choose the **Edit object** option or the **OLE Object activation** option.

What kinds of Objects can you add? An object is any piece of information created with an application for Windows that supports OLE. It is something that can be manipulated as a unit. An object can be all the information in a document, such as an entire word processor document, part of the information in a document, such as a phrase, or it can be an application,

Windows 9x already comes with applications that have some support for OLE (Wordpad and Paint for instance). Using Wordpad in combination with KeyGO, you can assign documents (or part of document) to any key or sequence of keys (using deadkeys). With Paintbrush you can, for instance, assign a PCX picture containing your logo to a key.

To use OLE features, you must install the OLE application you use with its appropriate Setup program.

Embedding is one of the two ways to add information from other applications. When you embed an object on a key, KeyGO stores the object in your KeyGO file. After you embed the object by using the **Insert New Object** (seep page 41 for more details) command in the **Edit** pull-down menu, you can start the source application (the application you used to create the object) and make changes to the object directly from KeyGO by double-clicking on the object. For example, suppose you create a Paintbrush picture and embed it in KeyGO. Once it is embedded, you can double-click the object to start Paintbrush and then make changes. To come back to KeyGO, click outside of the editing region

or exit the application. **To assign the object to a key**, drag-and-drop it to the desired key with the **Shift key depressed**. Because of the extent of the object frame, it is possible that some keys will not accept the assignment. If that occurs, drag the object to another key, and from there to the desired key. You can now press the corresponding key to have the OLE object pasted in or activated. To delete an object depress the Del (or Ctrl-X) key when the object is highlighted

Assignments can be done on all the different keyboard states (using Ctrl, Shift and Alt keys) and on any level of deadkeys and toggle keys as well.

An assigned OLE object will be represented on the floating keyboard by a red keycap.

Another way to embed an object to a key is by using the Cut-and-Paste features of Windows. After you've created an object from a OLE compliant Windows application, you can copy the entire object or part of the object using the **Edit copy** command of the source application and the **Edit paste** command of KeyGO. Select the object in the source application using the proper functions (refer to the manual of your application) then copy this object (normally by using the Edit, Copy command) to the clipboard. Then go back to KeyGO and paste the object using the Edit, Paste command. The object is now embedded to KeyGO. You can now assign this object to any desired key by dragging-and-dropping it **with the Shift key depressed**. Pressing this key from an OLE Windows application will paste this object directly in your document with the reference to the source application.

A third way to assign an object to a key is to drag-and-drop the object from the source application directly on a key in KEYGO. To do this, both the source application and the KeyGO builder with the proper keyboard layout have to be visible on your screen (use the restore and sizing functions of Windows to position the different applications on the screen). Select the object in your source application using the left mouse button and drag it to the proper key in KeyGO, then release the mouse button. If you drop the object on the keyboard layout background (white area) the object will not be assigned to a key. It will float over the keyboard layout waiting for you to assign it to a key. To assign it to a key, drag and drop it to the desired key with the shift key depressed. The advantage of having the object not yet assigned to a key is to give you the time to switch the level of the keyboard (Shift and deadkeys) where the object should be assigned.

You can change how an object will be pasted to an application by using the **OLE transfer Options** from the **Options** menu (see page 52).

The AltGr key

The AltGr key state is located on the right side of the Space Bar on most European keyboard templates. This key lets you have access to another state of keyboard, and therefore to new characters. In fact this state of keyboard is a combination of the Ctrl and the Alt keys (AltGr = Alt+Ctrl). If you are using a European type keyboard driver under Windows (see your Windows manual) you can assign characters and strings under the AltGr state by assigning them under a combination of Ctrl and Alt key states.

Example: To assign the ä character on the AltGr+a key combination, you would have to click on the Alt and the Ctrl keys on the keyboard layout of KeyGO and drag the ä character from the map to the a keycap.

Note: The AltGr key will work only if the current Windows keyboard driver supports this feature (Ex.: French under Keyboard of the control panel).

The Num Lock and Scroll lock keys

You can make any assignments to the Num Lock and the Scroll lock keys, as with other keycaps. However, even with something assigned on them, both of these keys will yield their normal

functions. For instance, you might decide to make the Num lock key a KeyGO toggle key. When depressed, the KeyGO keyboard state will switch to the proper assignment list under the toggle key AND the state of the numpad area will switch to numeric with the Num lock light turned on. The reason for this is to ensure that any function of these two keys defined by the current Windows keyboard driver will be kept. Also, it is much easier to follow the stage of the keyboard toggle state if the lights of these two keys are switched on/off during usage.

Deadkeys List
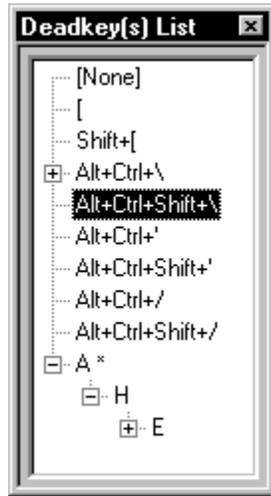
The Deadkey list is useful if you often use different levels of deadkeys (see below) or toggle keys. In essence, this tree-like list has the same functionality as the deadkey pull-down list on the toolbar, but with an easier interface for complex deadkey structures. To activate a deadkey or a toggle key level just click on the appropriate letter. If you want to expand or retract a deadkey branch; click on the plus or minus sign in front of the parent of the branch structure. The size of the Deadkey list box can be changed be dragging and dropping its border.

NOTE: Toggle key assignments appear in the deadkey list followed by a
        * character.

Deadkeys Overview

Deadkeys produce no output. They alter the behavior of the keys that follow it. Deadkeys are commonly used to produce diacritics (accents) on standard characters.

KeyGO also lets you define multiple levels of deadkeys for phonetic or multiple glyphs languages. Deadkeys are shown on the keyboard with a small rectangle at the bottom of the key.

CAUTION: The process of making single and multiple level deadkeys in KeyGO is very simple,
        but it can also become very confusing if the process is not completely understood.
        Read the following two chapters before attempting to make deadkeys.

Single level deadkeys

Creating single level deadkeys

To make single level deadkeys, first choose all the keys that you want to use as deadkeys. You can also use a combination of state keys and normal keys. For example, the combination of ALT+CTRL+SHIFT+(a letter) can be assigned as ONE deadkey.

Once you have chosen your deadkey, click on the Deadkey Icon , or choose **Make Deadkey** from the **Tools** menu, or use the Context menu using the right mouse button over the desired keycap.

You will now see that all the selected keys are now listed in the deadkey pull-down box on the

toolbar .
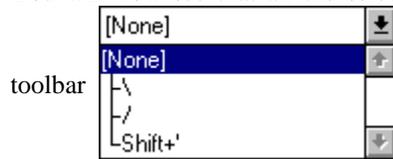
From this pull-down box, select the deadkey you want to build on. At this point, the rectangle on your deadkey will disappear. You will now be able to assign the characters or the string to be generated with this deadkey. To do so, drag-and-drop the object you need from the proper source (character map, string box, Explorer like application or an OLE 2.0 compliant application) to the proper key on the keyboard layout.

Caution: When you go back to making assignments without deadkeys, make sure to select "(Level 0)" from the deadkey pull-down list box. You also must select "(Level 0)" before making new single level deadkeys.

Deleting a single level deadkey

To delete a single level deadkey, you must first select the (Level 0) in the deadkey pull-down list. Then select the deadkey you want to unassign from the keyboard layout (the small rectangle at the

bottom of the key will now be visible). Finally, click **Unassign** from the **Tools** menu or the icon on the toolbar.

Another swift way to unassign a key is to drag-and-drop the key from the keyboard layout to the background of the layout (the white area), or use the Context menu (see page 17)

Note that all assignments built onto that deadkey will be lost.

Example

To assign the è character to a key combination of ' and e.

1. Create a deadkey with the ' character: Click the character ' on the keyboard layout then click **Make Deadkey** in the **Edit** menu (or use the icon on the Toolbar).

2. From the deadkey pull-down list on the toolbar select the ' character. (All new character assignments made at this point will be sent to your application only if you press the ' key first. The ' key does nothing by itself).

3. Drag-and-drop the é character from the character map to the e key on the keyboard layout.

4. From here it's simple to assign all the accent characters (è, ì, ù, à, etc.) to whichever keys you see fit.

5. Select "(Level 0)" from the deadkey pull-down box to ensure that unwanted assignments will not be made to the ' deadkey. Et voilà!

Multiple level deadkeys

Multiple level deadkeys are used primarily for phonetic typing of multiple glyphs languages. They allow you to press a sequence of keystrokes before producing the character (or string) you want in your application. For instance, if you occasionally use a Katakana Chinese font, you can generate one Katakana symbol with a keystroke sequence that represents the sound.

Creating multiple level deadkeys

1. To make your first multiple level deadkey, you need to begin by making a first deadkey, as described in the Single level deadkey chapter.

2. Select that deadkey from the deadkey pull-down list on the toolbar or from the Deadkey(s) list box activated by choosing the **Deadkeys list** option in the **View** pull-down menu..

3. Select the next deadkey in your sequence from the keyboard layout.

4. Press Make Deadkey .

5. (**Caution:** do not forget to Unselect all but the deadkey you are creating at this level.)

6. To continue your sequence of deadkeys, go back to step 2 and repeat. You can create an unlimited levels of deadkeys.

7. Now that your deadkey sequence is created, you can assign your object. Go to the pull-down list and select the last deadkey you created. You can then drag-and-drop your character (string, file or OLE 2.0 object) from the source (character map, the custom string table, Explorer like application or any OLE 2.0 compliant application) onto the key that will trigger the output.

If the deadkey pull-down list is set to (Level 0) all the keys are made single level deadkeys.

Note: It is very useful when testing multiple levels of deadkeys to have the floating keyboard activated. The keyboard will dynamically show you the different keyboard states when going through the different levels of the deadkey branch.

Deleting a multiple level deadkey

To remove a deadkey level from a multi-level deadkey branch, you must select from the deadkey pull-down list the deadkey level <u>above</u> the one you want to remove. Select from the keyboard layout the deadkey you want to remove (there will be a small rectangle at the bottom of the key), then click

Unassign from the **Tools menu** or use the unassign icon .

To remove a complete multiple level deadkey branch, set the deadkey pull-down list to (Level 0). You can then select on the keyboard layout the first deadkey of the sequence that you want to

remove and click the unassign icon .

You can also unassignme by dragging and dropping the key containing the deadkey from the keycap to layout background(the white area), or by using the context menu accessed by the clicking of the right mouse button over the proper keycap

Example

Let's say you want to assign the ☒ character from the Wingdings character set to a keystroke sequence made of S,T,O,P and ★ to a keystroke sequence of S, T, A, R .

It is recommended that you use the Ctrl and Alt state keys if you do not want your deadkey to interfere with normal typing.

1. Click on the Ctrl and Alt keys on the character map in order to put the keyboard in its "altered state".

2. Choose the S key on the keyboard layout and press the Make Deadkey on the Toolbar (or the **Make Deadkey** option from the **Edit** menu).

3. From the deadkey pull-down list on the Toolbar, choose the Alt+Ctrl+S branch.

4. Select the T character on the keyboard (be sure to Unselect the S character) and make it a deadkey.

5. Select the Alt+Ctrl+T branch from the deadkey pull-down list on the toolbar.

6. Select the O character on the keyboard (be sure to Unselect the T character) and make it a deadkey.
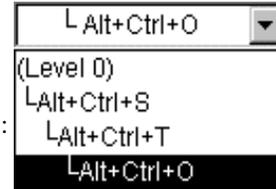
7. Look at the deadkey pull-down list. It should look like this:

8. Look at the Deadkeys list box (activated by selecting the **Deadkeys list** option in the **View** pull-down menu). It should look like this: You can Click the plus
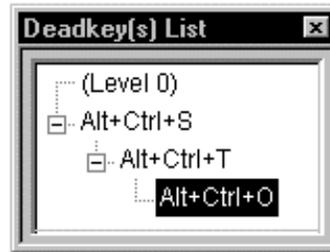
    signs (+) to display more deadkey branches or the minus (-) to contract a deadkey branch (see Deadkeys list box page   ).

9. Select the Alt+Ctrl+O character from the deadkey pull-down list on the toolbar or from the Deadkeys list box.

10. Choose the Wingdings font from the font pull-down list in the character map (see page 12) and drag the ☒ character to the P key on the keyboard layout. Notice that the P character was not made as a deadkey, since it is the last character in the keystroke sequence that will trigger the ☒ symbol.

11. Every time you press Ctrl + Alt and S, T, O, P in sequence, the ☒ will be sent to your application, providing of course that the font used in your application is set to the Wingdings character set and that KeyGO is activated (see page 44). If you are using the floating floating keyboard option (see page 46) notice how it is reacting. At each keystroke it will show the next KeyGO assignment key choice.

12. You can now assign the ★ symbol to Ctrl + Alt and S, T, A, R. You already have as deadkey the Alt+Ctrl+S above Alt+Ctrl+T in the deadkey pull-down list. Select the Alt+Ctrl+T from the deadkey pull-down list.

13. Select the A character on the keyboard (Alt and Ctrl should still be selected) and make it a deadkey.

14. Select Alt+Ctrl+A from the deadkey pull-down list or from the Deadkeys list box.

15. Choose the Wingdings font from the font pull-down list in the character map, and drag the ★ character to the R key on the keyboard layout. Notice that the R character was not made as a deadkey, since it is the last character in the keystroke sequence that will trigger the ★ symbol.

16. Your pull-down list should look like this:

```
[None]
 └Alt+Ctrl+S
   └Alt+Ctrl+T
     ├Alt+Ctrl+O
     └Alt+Ctrl+A
```
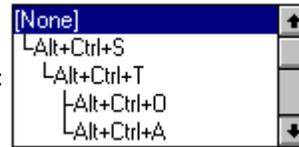
Caution: Make certain you select the correct branch each time a new deadkey level is added.

Toggle keys Overview

Toggle keys produce no output. They alter the behavior of the keys that are subsequently pressed. Toggle keys are used to switch the whole keyboard into another state. In essence toggle keys are the same things as deadkeys, but need to be depressed a second time to bring back the original keyboard state. They act like the Scroll lock, Num lock and Caps Lock, but act on the whole keyboard.

Toggle keys can be used as an alternative to using of multiple keyboards. For instance a whole keyboard can be built under a Ctrl-1 toggle key, another one under a Ctrl-2 combination, etc.

The Scroll Lock and Num Lock keys can be completely redefined by making them toggle keys.

KeyGO will NOT let you define multiple levels of toggle keys (it would be much too confusing). However it is still possible to mix toggle keys and deadkeys on different levels. In other words, toggle keys can have an unlimited number of deadkeys under them, and a deadkey sequence can contain one and only one toggle key in its branch. Toggle keys are shown on the keyboard template and on the floating keyboard with a small crossed rectangle at the bottom of the key when they are in their first state, or with a small red rectangle when they are in their second state. Both states will be shown as a red rectangle on the floating keyboard.

CAUTION: The process of making single and multiple level of deadkeys containing toggle keys in KeyGO is VERY simple, but it can also become very confusing if the process is not completely understood. (See Deadkeys Overview page 25).

To make toggle keys, first choose all the keys that you want to use as toggle keys (normally one). You can also use a combination of state keys and normal keys. For example, the combination of ALT+CTRL+SHIFT+a letter can be assigned as ONE toggle key. Once you have chosen your toggle keys, choose **Make toggle** from the **Tools** menu, or by right clicking on the button mouse. You will now see that all the selected keys are listed in the deadkey-toggle key pull-down box on the toolbar and in the deadkey-toggle key list floating box. Toggle keys are always shown in these boxes followed by a * character.

From this pull-down box, select the toggle key you want to build on. At this point, the crossed rectangle on your toggle key will change to a red rectangle. No assignment can be made on the toggle key, when it is in the second state. You will now be able to assign all the characters, string or deadkeys to be generated under the second state of the toggle key. To do so, drag-and-drop the characters or strings you need from the character map or string box to the proper key on the keyboard layout or make a deadkey to any desired key.

Caution: When you want to go back making normal assignments at the first state of the toggle key, make sure to select the first level of the toggle key from the deadkey-toggle key pull-down list box-normally "(Level 0)".

Combining deadkeys or toggle keys with string assignments

You can combine String Assignments (see page 18) with Deadkeys (see page 22). This is useful in producing a string of characters in your application (word processor, communication program, etc.) with a keystroke sequence.

These combinations are performed the same way as simple character assignments with deadkeys. The only difference is that string assignments must be built beforehand, and drag-and-dropped from the custom string table (instead of the character map).

Example: Let's construct a process that will send the *GDG Systèmes, Inc.* mailing address to the Word Processor if the letters G, D, G are pressed in sequence.

1. Define multi-level deadkeys with G and D. Click on the G key on the keyboard and make it a deadkey using the Deadkey icon on the toolbar ![icon], or choose **Make Deadkey** from the **Tools** menu or using the Context menu (see page 17).

2. Choose G from the pull-down deadkey list on the toolbar. Unselect G from the keyboard and select D, and make it a deadkey as you did for G (step 1). Your deadkey sequence is now finished.

3. You will now need to build a string. From the custom string table choose **Add**. In the Name dialog box, enter GDG. In the **Value** dialog box, enter the following string: GDG Systèmes, Inc. {enter}110, McGill Street Suite 202{enter}Montreal, Quebec{enter}H2Y 2E5. Click **OK**. Note: All special commands need to be entered enclosed in { } (see page 18).

4. Select D in the deadkey pull-down list on the toolbar, and drag the GDG string name from the Custom String table to the letter G on the keyboard layout.

5. Activate your keyboard by using the toolbar icon ![icon] or the Keyboard menu. From now on, each time you press in sequence the letters G, D, G, the full GDG address will be sent to your application.

If you are using the floating keyboard option (see page 46) note how it is reacting with each keystroke. Every time a key is pressed, the next possible KeyGO assignments are shown in light blue (with a rectangle for the deadkey, and an X for the strings).

Combining deadkeys or toggle keys with file or OLE assignments

You can combine File Assignments (see page 22) and OLE 2.0 objects (see page 23) with Deadkeys (see page 22). This can be useful to launch an application, or to paste an OLE object into your current application with a keystroke sequence.

These combinations are performed the same way as simple character assignments with deadkeys.

Example of a file Assignment: Let's construct a process that will launch Winword if the letters Ctrl W, O, R, D are pressed in sequence.

1. First build the multi-level deadkeys with Ctrl W, O and R: Click on the Ctrl key and the W key on the keyboard, and make it a deadkey using the Deadkey icon on the toolbar ![icon], the **Make Deadkey** from the **Tools** menu or use the Context menu (see page 17).

2. Choose W from the pull-down deadkey list on the toolbar. Unselect W from the keyboard and select O, and make it a deadkey as you did for W (step 1).

3. Choose O from the pull-down deadkey list on the toolbar. Unselect O from the keyboard and select R, and make it a deadkey as you did for W (step 1). Your deadkey sequence is now finished.

4. Select R in the deadkey pull-down list on the toolbar, and drag the Winword program from Windows Explorer to the letter D on the keyboard layout. You can also Right click on the D key and select the **Add/Edit File assignment** option. Select the Winword program from the proper directory (normally C:\Office\Winword) and validate.

5. Activate your keyboard by using the toolbar icon 🟢 or the Keyboard menu. From now on, each time you press in sequence the letters Ctrl W, O, R, D Winword will be launched.

Note that it would also be possible to assign a Winword document file instead of the program itself. Doing so would launch Winword with the proper document ready for editing.

If the Floating keyboard is visible (see page 46), note how it is reacting with each keystroke. Every time a key is pressed, the next possible KeyGO assignments are shown in light blue (with a rectangle for the deadkey, and an blue rectangle for the file assignment).

The same kind of process can be used to assign an OLE object to a sequence of keys. For instance, Step 4 in the last example could be replaced by:

4. Select R in the deadkey pull-down list on the toolbar, and drag the text from Winword to the letter D on the keyboard layout (or use the Copy and Paste feature of Windows between Winword and KeyGO). You can also select the **Insert new Object** option from the **Edit** menu. If the **Insert new Object** or the **Copy and Paste** command are used to get the object, it needs to be assigned to the R key by dragging and dropping the object to the R key <u>with the shift key depressed</u>.

Multilanguage support

KeyGO 2.x can automatically switch between script/fonts if your application has multilanguage support (Word 7.0, Wordpad. Etc.). For this feature to work you have to make sure that the **Automatic script switching** option is enabled under the **Keyboard Language Options** dialog box in KeyGO's **Options** pull-down menu.

IMPORTANT: The language features will only work with Windows 9x and Windows NT version 4.0. The language related options in KeyGO will be disabled or absent when running under version 3.5x of NT.

IMPORTANT: For the multilanguage features of KeyGO to work properly, including Greek, Cyrillic and Central European fonts, you need to install Multilanguage support under Windows 9x. This is done by opening the **Control Panel**. Choose the **Add/Remove program** icon, and from the Windows Setup tab, validate the Multilanguage support option. You also need to install the Greek, the Russian and a Central European keyboard from the Keyboard option (language) in the Control Panel.

IMPORTANT: When assigning a Western type character to a key, the language reference (and keyboard) that will be sent to the application will be the first one found in the list of keyboard in the Keyboard options of the control panel that matches the Western script. If you want this to be different, choose a new Western keyboard template in the keyboard options of Windows as the default one.

For some programs like Word 7.0 it is necessary to have the keyboard language preloaded to use the script switching features of KeyGO. Some programs, like Wordpad will switch fonts regardless of the keyboard loaded, and will even let you automatically switch to some font like Symbol. To load a new language under Windows 9x, open your Control Panel, choose Keyboard, Language, Add, and from the list choose the corresponding language keyboard and <u>restart</u> KeyGO.

For example, if you use a Greek font in your application, a keyboard language layout for Greek has to exist in the Windows 9x environment.

This feature will not only switch the script of the font but also the language in which you are writing. This way you can take full advantage of the language features of your main application (spelling, format, etc.)

The language for the keyboard can also be dynamically changed in KeyGO. The language of a keyboard can be selected from the System tray of Windows 9x, or better, from the **Tools, Language** pull-down menu in KeyGO. The languages that will appear in these menus are only the preloaded ones (by Windows 9x. See above). The keyboard layout will change its appearance to reflect the newly selected language (even for non-roman languages like Greek or Russian). If the option **Use KeyGO language layout** is set in the Keyboard Language Options dialog box, every time an <u>unassigned</u> key is depressed, a language switching command corresponding to the one used to lay out the keyboard template will be sent to the application. In essence, KeyGO will force your application to switch to the language specified by the layout of your keyboard, replacing all Windows 9x defaults.

Keyboard Assignments

This window will come up when the **Add** or the **Modify** button from the **Keyboard dialog box from the Options pull-down** menu (see page 46) is pressed. It will let you build the list containing the keyboard layouts. You should include in this list all the keyboard layouts that you will want to access dynamically from any Windows application using a hotkey.

File: This dialog box lets you enter the name of the KeyGO file that will be added to the File Mappings list in the Options menu. The **browse** button lets you find the desired file from the disk.

Hotkey: The hotkey is used to rapidly turn the KeyGO control to the keyboard layout specified in the File dialog box. It can be used in all Windows applications. To enter a hotkey, just click in this dialog box and press the combination of keys you want to use to access the current selected keyboard layout file. It is highly recommended to use Ctrl, Alt and Shift key combinations. However, any key or state key combination can be used, with the exception of some Alt-key combinations (systems commands). If your key combination does not appear in this box when pressing it, choose another one

The **OK** button lets you validate your choice in the File mapping list of the Options menu.

It is possible to make KeyGO send a string each time a keyboard layout is activated using a hotkey. This is useful if you want an application to do certain things each time a new keyboard is loaded. For instance, it can be useful to have KeyGO send a string to the word processor that will shift the font selection to the one used to design the keyboard layout.

To do this enter a string (as describe in the Strings chapter page 18) in the **Auto activate** dialog box in the **Keyboard General Options** from the **Options** pull-down menu.

Note: To be able to reactivate the keyboard layout specified in the Startup File, this file must also be listed in the File Mapping List.

Floating Action Bar

When the optional Floating Action Bar is validated in the View pull-down menu and **the View Preferences** dialog box in the **Options** pull-down menu, the following Action Bar will always be visible from all Windows applications.

The ⬤ icon is a toggle button that lets you turn KeyGO On and Off. (note: the floating keyboard will not be visible if KeyGo is off).

The ⌨ icon is a toggle button that lets you turn the floating keyboard On and Off.

The 🗐 icon lets you have access to the list of available keyboard layouts as entered in the Keyboard list in the Options dialog box with their corresponding hotkeys. Selecting a keyboard from this pull-down menu is the equivalent of loading it with the hotkey.

It will make the chosen keyboard active and turn KeyGO on. The name of the keyboard file will be shown, using its title if it exists. The title for a keyboard file can be entered in the **Keyboard Summary** dialog box from the Options pull-down menu. **Use title, if it exists, to list keyboard in menu** option from the **View Preferences** dialog box also have to be checked for this to work.

The ✖ is a toggle key that lets you activate or put KeyGO's builder back in the background. If a swift change in your keyboard layout is necessary, click on this button to activate the builder. Once your modifications are made, you can click on this icon again to put KeyGO back to the background.

Windows 9x System Tray

Turn KeyGO On and Off

List of available keyboard layouts

Turn the floating keyboard On and Off

Activate KeyGO's builder
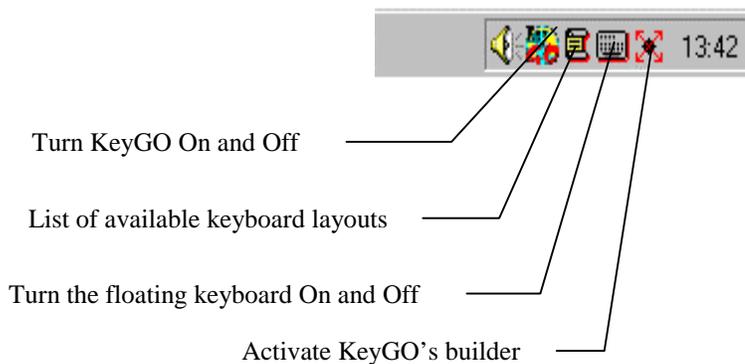
When the optional **Use Windows 9x or NT System Tray** option is validated in the **View Preferences** dialog box in the **Options** pull-down menu, KeyGO access icons will be shown on the

Windows 9x System tray. In essence these icons are the same found on the Action Bar (see above) but they are anchored in the System tray of Windows 9x or Windows NT. You can select from the View Preference dialog box which icon will be show (see page 51).

Building Windows 9x keyboard driver

You can use KeyGO to generate Windows 9x keyboard drivers.

This possibility exists only when using KeyGO under Windows 9x. The **Build Keyboard Driver** option will not be available under Windows NT 3.51 or Windows NT.

When selecting the **Build Keyboard Driver** from the **File** pull-down menu, a dialog box will come up. This box will let you choose different options for generating drivers (see page 38 for details on these options).

The first thing to do before converting a keyboard layout to a driver is to choose the keyboard language of the layout (**Language** from the **Tools** pull-down menu). This is important for two reason; First, the generated driver will inherit the local language identifier of the current keyboard. Second, all default assignments of the current Windows keyboard layout (including deadkeys) will be preserved in the newly generated driver. **Always document a new driver** in order to keep track of your keyboard layouts. You can use the KeyGO print command to do so.

There are some limitations on using a generated Windows keyboard driver instead of using KeyGO in runtime mode. Keep in mind, when building a keyboard for conversion, that only assignments normally known to work under Windows will be allowed. The keyboard driver will only permit character assignments and one level of deadkeys on any level of state key. However, all KeyGO keyboard layouts can be used to generate a driver. Assignments that are not permitted will just be disregarded.

Here is a list of limitations, and a table of permitted character assignments:

Limitations

- String assignments will not be converted.

- OLE and File assignments will not be converted.

- Only the first level of a multiple deadkey branch will be converted.

- Characters assigned to direction keys will not be converted (numeric keypad assignments will, however, be converted) .

- Application shortcuts will override any character assignments. Ex.: Assigning under Ctrl-S will not change the Save function under most Windows applications.

- Assignments under the Alt key will not be converted.

- Assignements under the Ctrl, Alt and AltGr state keys (or in any combination) under a second level of a deadkey will not be converted. They should be done on a unshifted key or on a shifted key (see the rule table below).

- Toggle keys will be converted to deadkeys.

- When the driver is used in KeyGO as a keyboard language layout, assignments done on function keys will not be shown on the keycaps. They nonetheless are functional.

Installing the driver

Here are the steps you should use to install the newly created driver in Windows:

- Build the driver in KeyGO with the **Add the layout name to Windows keyboard list** option checked.

- Open Windows' Control Panel and choose **Keyboard**.

- Select the **Language** tab.

- From here you can select the language on which you want to attach the new driver, you can use **Add...** to install a new one.

- Click the **Properties** button. Then, from the **Keyboard layout** pull-down menu, choose the name of your keyboard driver as it was entered in the **Title in the control panel property** edit box of the **Build keyboard Driver** dialog box in KeyGO.
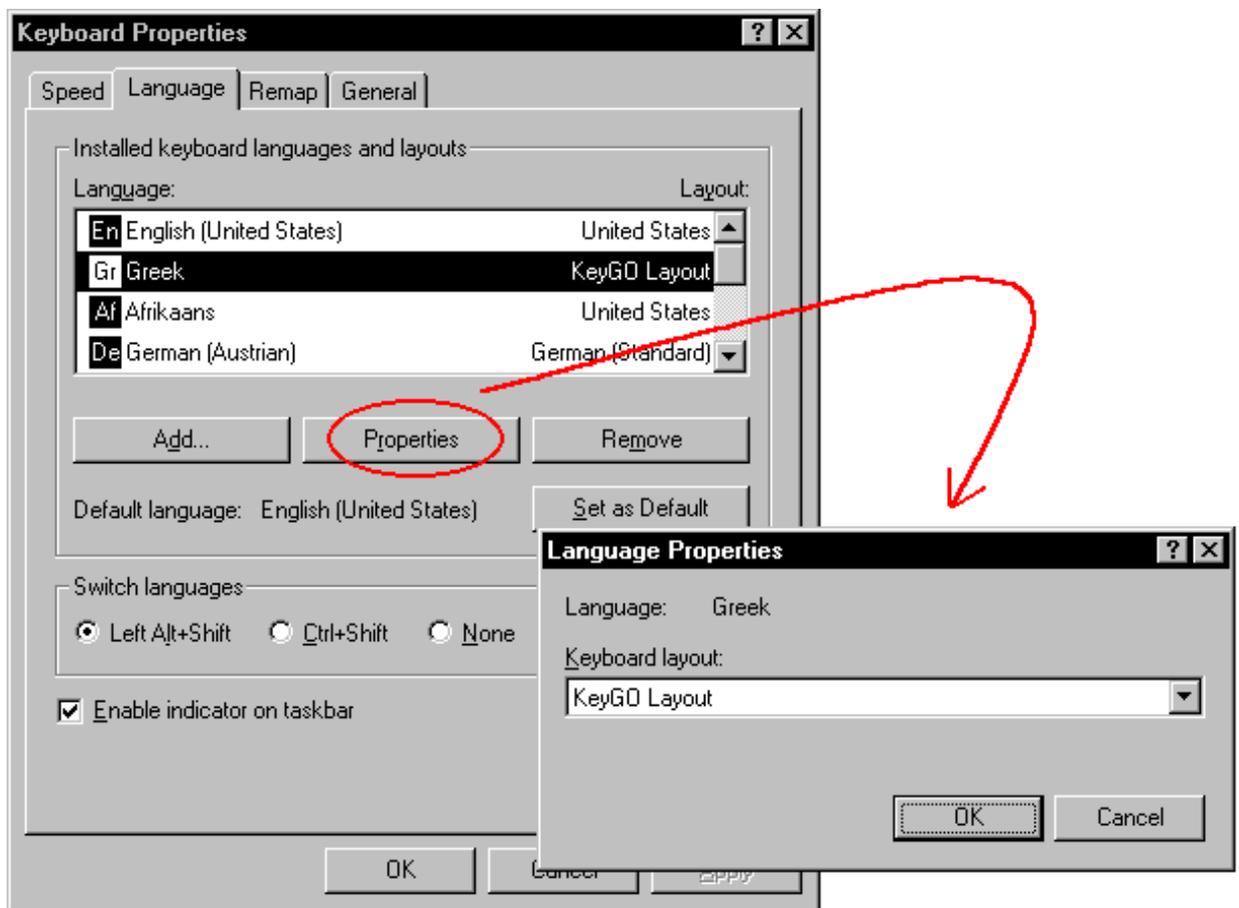
From now on, your layout will be active each time the language associated with your driver is selected within a given application.

To make your driver the default one, you should associate it with the default language of Windows, or make the language attached to the driver the default. To do so, choose the proper language in the **Keyboard language** option of the Control Panel and click on **Set as Default**.

You do not have to reboot Windows to make a new driver active. You should, however, restart Windows if a new driver was built over an existing one already in use by Windows.

KeyGO does not have to be running for a generated keyboard driver to work.

If the **Add the layout name to Windows keyboard list** is not checked, a keyboard driver will be generated, but the name will not be added to the available keyboard layout in the Keyboard option of the Control Panel. To install these drivers you need the KBDINST program, provided to site licensees, or OEM customers who have the rights to distribute and copy these drivers.

Permission table

✓ - The combination is permitted

○ - The combination is <u>not</u> permitted

☑ - The combination is permitted with limitations. The Windows shortcuts will override them.

	Key combinaison								
	Unshifted	Shift	Ctrl	Alt	AltGr*	Ctrl-Shift	Alt-Shift	AltGr*-Shift	Note
First level	✓	✓	☑	○	✓	☑	○	✓	Not on directional keys
Second level of deadkey	✓	✓	○	○	○	○	○	○	Only on non-function keys

*AltGr = Alt+Ctrl key.

Note on deadkeys

All the default deadkeys defined in the original keyboard language layout of Windows will be transferred to the new driver.

Only the first level of deadkeys will be converted. You can only define a maximum of 17 deadkeys in any state level of your layout, including the one already defined by the Windows keyboard language. If you want to add new assignments under an existing deadkey, first select the original deadkey under which new assignements will be done, and make it a KeyGO deadkey (see page 25). Select this deadkey from the deadkey pull-down list and make your new assignments. All the original assignments will be preserved and the new assignments will be added. Even if a deadkey can be defined on any keyboard keycap, assignments on the second level of the deadkey can only be made on <u>the non-function keycaps</u> (normal character keys). Ex.: Shift-F2 can be a deadkey, but cannot support a second level of a deadkey assignment.

IMPORTANT: Some Windows keyboard drivers have deadkey definitions that KeyGO will not convert. If you encounter such Windows drivers, you can either choose another one (from the Language pull-down menu) and make the proper change to the deadkeys, or build the new deadkeys from scratch using a plain keyboard driver (ex.: US keyboard).

Menu Commands

New (File menu)

Use this command to create a new keyboard layout in KeyGO.

You can open an existing keyboard layout using the Open command.

Shortcuts

Toolbar:

Keys: CTRL+N

Open (File menu)

Use this command to open an existing keyboard layout in a new window. You can have several keyboards open at once. Use the Window menu to switch among the multiple open keyboard layouts. See Window 1, 2, ... command.

You can create new keyboard layouts with the New command.

Shortcuts

Toolbar:

Keys: CTRL+O

From the Open dialog box, the following options will allow you to specify which file to open:

File Name

Type or select the filename you want to open. This box lists files with the .KGO extension.

Files of Type

Select the type of file you want to open (files with the .KGO extension or all files).

Look in

Select the drive and directory in which KeyGO stores the file that you want to open.

Close command (File menu)

Use this command to close all windows containing the active keyboard layout. KeyGO suggests that you save changes to your keyboard layout before you close it. If you close a keyboard layout without saving, you lose all changes made since the last time you saved it. Before closing an untitled keyboard layout, KeyGO displays the Save As dialog box and suggests that you name and save the keyboard layout.

You can also close a keyboard layout by using the Close icon on the keyboard layout window to the right of the keyboard layout title bar.

Save (File menu)

Use this command to save the active keyboard template. When you save a keyboard layout for the first time, KeyGO displays the Save As dialog box so you can name your keyboard layout. If you want to change the name and directory of an existing keyboard layout before you save it, choose the Save As command.

Shortcuts

Toolbar: 🖫

Keys: CTRL+S

Save As... (File menu)

Use this command to save and name the active keyboard layout. KeyGO displays the Save As dialog box so you can name your keyboard layout.

To save a keyboard layout with its existing name and directory, use the Save command.

From the Save As... dialog box, the following options allow you to specify the name and location of the file you are about to save:

File Name

Type a new filename to save a keyboard layout with a different name. A filename can contain up to eight characters and an extension of up to three characters.

Save as Type

Select the type of file you want to save to (files with the .KGO extension or all files).

Save in

Select the drive and directory in which you want to store the keyboard layout.

Build Keyboard Driver... (File menu)

This option only exists when using KeyGO under Windows 9x. The **Build Keyboard Driver** option <u>will not</u> work under Windows NT.

See page 34 for more detail on the limitations and rules of building a Windows 9x keyboard driver, and how to activate it under Windows.

When selecting this option, the following dialog box will come up:

KeyGO driver builder	✕

KeyGO keyboard layout:

`Cndmulti`

[Generate Driver]

Title in the control panel property:

`KeyGO Layout`

[Cancel]

Driver name:

`KeyGO`

[Help]

☑ Add the layout name to Windows keyboard list

AltGr key usage
- ◉ Use keyboard default
- ○ Activate AltGr key
- ○ Deactivate AltGr key

CapsLock key usage
- ◉ Use keyboard default
- ○ Toggle CapsLock
- ○ Shift Lock CapsLock

Consult Help for instructions and rules on building a Windows keyboard driver

The **KeyGO keyboard layout** box shows you the file name of the KeyGO keyboard layout that will be converted to a native Windows 9x keyboard driver.

By default the **Title in the control panel property** edit box will be *KeyGO Layout*, and should be changed to any description you see fit. This description will be used to list the new driver in the Control Panel of Windows.

You should enter in the **Driver name** edit box the name of the keyboard driver <u>file</u> that will be generated by KeyGO. This name can only have a maximum of 8 characters, and the extension (.KBD) will be automatically added by KeyGO. No path, drive or extension should be entered in this edit box if the **Add the layout name to Windows keyboard list** is checked.

If the **Add the layout name to Windows keyboard list** is checked, the new keyboard driver will be constructed in the Windows 9x system directory, and the title of the driver will be added to the keyboard layout list in the Windows 9x control panel. See page 34 for details on how to access your keyboard driver from Windows. If unchecked, only the keyboard driver file will be generated to the drive and path specified in the **Driver name** edit box.

The **AltGr key usage** multiple choice box lets you choose how the newly generated driver will make use of the right Alt key (see page 24 on the AltGr key). If **Use keyboard default** is selected, the right Alt key will react in accordance with the current selected keyboard language. The **Activate AltGr key** choice will force the keyboard driver to make use of the AltGr key. **Deactivate AltGr key** will disable the AltGr key. The assignments done under the Ctrl-Alt(-Shift) key combinations will be available even if the **Deactivate AltGr key** is selected. They will not, however, be accessible using the right Alt key, but only with the Alt-Ctrl(-Shift) key combination.

The **CapsLock key usage** multiple choice box lets you choose how the newly generated driver will turn its CapsLock key off. If **Use keyboard default** is selected the CapsLock key will react in accordance with the currently selected keyboard language. If **Toggle CapsLock** is selected, the CapsLock will be a toggle key. If **Shift Lock CapsLock** is your choice, the CapsLock will be turned off by depressing the Shift key.

You may want to use these last two options to change only the aspect of a given Windows keyboard driver. For example, if you are using the Windows French keyboard layout, but you want to change the way the CapsLock key is turned off, create a new KeyGO keyboard layout (**File**, **New**). Then choose the French language from the **Language** pull-down menu from the **Tools** menu. Now choose the **Build Keyboard driver** option (without having made any assignments) and check the **Toggle CapsLock** option. The newly generated keyboard driver will have the exact characteristics of the French driver, but with the CapsLock keycap now acting as a toggle key.

Print command (File menu)

This submenu lets you print a list of key assignments for your current keyboard layout (see the Windows manual).

Print Preview command (File menu)

This submenu lets you preview the print-out list of the different key assignments (see the Windows manual).

Shortcuts

Toolbar:

Printer Setup command (File menu)

This menu lets you choose and configure the printer to which the hard copies will be sent. (see the Windows manual).

Unlock KeyGO (File menu)

KeyGO will run in demo mode the first it is launch. In this mode you won't be able to generate native keyboard drivers and a dialog box will pop-up every 50 typed character, reminding you of the demo status of KeyGO. To unlock the release version; click on the file menu and choose *Unlock KeyGO*. Enter the key code that was provided to you by GDG Systems or your distributor and click OK.

1, 2, ... command (File menu)

Use the numbers and filenames listed at the bottom of the File menu to open the last four keyboard layouts you closed. Choose the number that corresponds to the keyboard layout you want to open.

Exit command (File menu)

Use this command to end your KeyGO session. You can also use the Close command on the application Control menu. KeyGO prompts you to save keyboard layouts with unsaved changes.

Shortcuts

Mouse: Double-click the application's Control menu button .

Keys: ALT+F4

Cut command (Edit menu)

This option lets you delete the currently selected embedded OLE object (see page 23) and copy it to the clipboard. This command will not cut an object assigned to a key. To remove an assignment, use the **Unassign** command (see page 45).

Copy command (Edit menu)

This option lets you copy the currently selected embedded OLE object (see page 23) to the clipboard. This command will not copy an object assigned to a key.

Paste command (Edit menu)

This option lets you paste the OLE object (see page 23) currently in the clipboard to the current keyboard layout. The object in the clipboard can be any kind of information copied from any Windows application that supports the copy command. When an object has been pasted to the keyboard, it can be assigned to a key by dragging and dropping it on the keycap with the shift key depressed (see OLE object page 23).

Insert New Object (Edit menu)

An object can also be pasted to the current keyboard layout by choosing **Insert New Object** from the **Edit** menu. A menu will come up asking for information regarding the object that should be pasted onto the keyboard layout (see OLE object page 23).
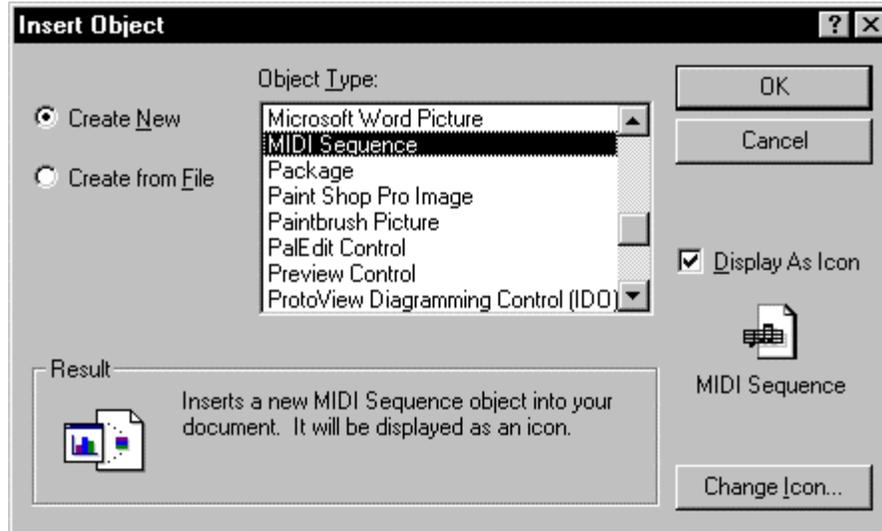
When in the Create New state, this dialog box will let you choose the application in which the object should be constructed. Clicking the OK button will launch the selected application and let you construct your object. Upon exiting the application KeyGO will be updated with the object not yet assigned to any key.

When the **Create from File** is selected, the object will be an existing file that you will have to select with the Browse button. This file can be embedded or linked to KeyGO. When embedded, the object (the file) will remain unchanged even if the original file is changed. If linked (by clicking on the Link box), the object will automatically be updated with the original content of the file each time the keyboard is loaded. A subsequent change to this file will be reflected in the object pasted onto the KeyGO keyboard layout. Pasting a file to a key using this method is NOT the same as assigning files to keys (see file assignments page 22). OLE assignments are principally done to paste documents into an existing one, and File assignments are done to launch documents or applications. (For more information on linking and embedding, see your Windows manual.)

By default, **the Display As Icon** box will be checked, but it is possible to show the pasted object by its content instead of by an icon. If you use this option, be sure to paste a small document, since keyGO will NOT resize the object to fit within its frame, making it impossible to move or edit. If

you edit an object (double-click) represented by its content, the original application will not be called as an external program, but will be embedded within KeyGO. KeyGO becomes the receptacle for all the functionalities of the original application. To get out of this state when all the editing is done, just click outside of the editing region.

New File assignment (Edit menu)

This command lets you assigne a file to a key. When chosen, a dialog box will let you choose the file that needs to be assigned to a key (see file assignments page 22).

When the file has been chosen it will be assigned to the first available key on the keyboard starting on the upper left corner of the layout. From there it can be moved to any desired key. It is a good idea to first select the level of deadkey or shift state before calling this option. The purpose of assigning files to keys is to be able to launch an application (equivalent to a shortcut key under Windows) or a document.

Launch Explorer command (Edit menu)

Use this command to launch the Windows files Explorer program. Under Windows NT 3.51 Windows File Manager will be launched instead.

This command is usefull if you want to make quick file assignments (see page 22) from Explorer or File Manager.

Toolbar command (View menu)

Use this command to display and hide the Toolbar (Alt,V, T). The Toolbar includes buttons for some of the most common commands in KeyGO, such as File Open. A check mark appears next to the menu item when the Toolbar is displayed. See also "Toolbar" (page 8) for more detailed explanations.

Status Bar command (View menu)

Use this command to display and hide the Status Bar. The Status Bar describes the keyboard latch state (Num Lock, Scroll Lock) and the action to be executed either by the selected menu item or by

depressing a toolbar button. check mark appears next to the menu item when the Status Bar is displayed.

See also Status bar (page 11) for more detailed explanations.

Character map command (View menu)

Use this command to display and hide the Character Map. This includes a font selection pull-down list and the complete ANSI Windows character set from which the drag-and-drop process takes place. A check mark appears next to the menu item when the Character Map is displayed.

See also "Character map" (page 12) for more detailed explanations.

Shortcuts

Toolbar:

String box command (View menu)

Use this command to display and hide the Custom String table, which includes a list of all available strings from which the drag-and-drop process takes place. A check mark appears next to the menu item when the Custom String table is displayed.

See also the chapter "String assignments" (page 18) for more detailed explanations.

Shortcut

Toolbar:

Deadkey list (View menu)

Use this command to display or hide the Deadkey(s) list box. This box has the same function as the deadkey pull-down list on the toolbar, but is easier to use for complex deadkey and toggle key levels (see page 25)

Shortcut

Toolbar:

Floating keyboard (View menu)
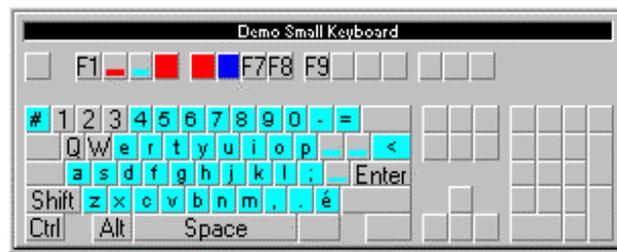
Click on the icon on the Toolbar, on the System tray or the Action bar if you want to have an Always-on-top floating keyboard representing the current layout. This floating keyboard will always stand in front of your application and will dynamically show the key assignments with the proper state key status. An X on a key represents a string assignment. An horizontal rectangle at the bottom of the key represents a deadkey, a horizontal red rectangle at the bottom of the key represents a

toggle key. A blue keycap represents a file assignment and a red keycap represents a OLE object assignment. When a deadkey, a toggle key or a state key (Ctrl-Shift and Alt) is pressed, the keyboard will change to show the next available assignments that are attached to this deadkey or state key level (strings, characters, or other deadkeys).

This keyboard can be moved by dragging and dropping from within its content anywhere on the screen. The caption of the title bar is the title of the active keyboard layout as entered in the **Keyboard Summary..** dialog box in the **Options** pull-down menu (see page 48).The size of this keyboard can also be changed by dragging and dropping its border or by changing the size of the original keyboard layout in the KeyGO builder. A double-click on the content of the floating keyboard will automatically activate the keyboard builder with the proper file loaded ready for editing.

Shortcuts

Toolbar, Action bar or System tray: 🖮

Action Bar command(View menu)

Use this command to display and hide the Always-on-top Action Bar. See page 33.

Scale to Fit command (View menu)

Use this option to scale the current keyboard layout to match the size of its surrounding window. (Note: The percentage pull-down menu will always be updated with the proper zoom information. Changing the percentage in this pull-down menu will automatically turn off the Scale to Fit option.)

Zoom command(View menu)

This menu lets you choose the zoom factor of the current keyboard from a preselected list. Choosing a new setting for the zoom will deactivate the **Scale to Fit** command (see above) and will be reflected on the toolbar's zoom pull-down menu.

Activate command (Tools menu)

Makes the current Keyboard layout the active one. The name field on the Status bar will be updated accordingly. The name of the keyboard that would activated through the use of this command is shown, on the same line, within braces.

Caution: If you select another Keyboard layout for editing, the active Keyboard layout remains the one previously chosen. To make the new layout active, you must press the Activate button again.

Shortcuts

Toolbar, Action Bar: 🟢

System tray: 🖼

Deactivate KeyGO command (Tools menu)

Unhooks KeyGO from Windows. Makes KeyGO inactive.

Shortcuts

Toolbar: ![x icon]

Action Bar: ![green circle icon]

System Tray: ![system tray icon]

Unassign command (Tools menu)

Use this command to unassign a character from a key. Select one or more keys. These should turn dark red. Choose Unassign from the Edit menu, or the ![icon] icon on the toolbar. The key is now in its original state.

Another and sometimes easier method is to drag-and-drop the character from the key to the keyboard layout background.

Shortcuts

Toolbar: ![toolbar icon]

Deselect All (Edit menu)

Use this command to deselect all the keys currently selected

Make deadkey (Tools menu)

Use this command to make the selected key a deadkey.

Deadkeys produce no output. They alter the behavior of the keys that follow it. Deadkeys are commonly used to produce diacritics (accents) on standard characters.

KeyGO also lets you define multiple levels of deadkeys for phonetic languages. Deadkeys are shown on the keyboard with a small rectangle at the bottom of the key.

CAUTION: The process of making single and multiple level deadkeys in KeyGO is very simple, but it can also become very confusing if the process is not completely understood. Read the chapters "Making single level deadkey" (page 22) and "Making multiple level deadkey" (page 26) before attempting to make deadkeys.

See also "Combining deadkeys and strings" (page 29).

Shortcuts

Toolbar: ![toolbar icon]

Make Toggle key (Tools menu)

Use this command to make the selected key a toggle key (see page 29).

Toggle keys produce no output. They alter the behavior of the keys that follow it. Toggle keys are state keys which act the same way as NumLock, ScrollLock or CapsLock keys.

CAUTION: The process of making single and multiple level toggle keys in KeyGO is very simple, but it can also become very confusing if the process is not completely understood. Read the chapters on toggle keys (page 29) before attempting to make them.

Copy Assignment to New keyboard command (Tools menu)

This option lets you create a new keyboard with the same assignments as the current one. This option is useful when creating a new keyboard based on another one when a change of keyboard template is necessary.

Note: No assignments are lost during a transfer from a 101/102 key keyboard to a 83/84 keys keyboard even if some of the keycaps are not present. This feature will let you reconstruct a 102 key keyboard from a 83/84 key keyboard with all of the original assignments.

New Font for All Character Assignments (Tools menu)

This option lets you change the font used by all assigned characters on the current keyboard layout. The font that will be used by this option is the current character map selected font. To use this feature, first select the font that you want to use from the **Character map** font pull-down menu (see page 12), then click on this option in the Tools menu.

This option is useful when a keyboard layout is used on another machine that does not have the same font name as the original one.

IMPORTANT: There is no Undo on this function. All the character assignments will use the newly chosen font under all deadkeys and state levels.

Language (Tools menu)

This menu lets you choose the language of the current keyboard layout. This list only contains the keyboard language preloaded by Windows. Choosing a new language changes only the original character assignments on the keyboard. None of the KeyGo assignments (characters, strings, OLE or files) will be changed when selecting a new language. However, it will change the way the keyboard layout and floating keyboard are displayed on the screen. Eventually it might also change the behavior of the target application if the **Use KeyGO language layout** option is set in **the Keyboard language option** from the **Options** pull-down menu. For more details see Multilanguage support page 31

Startup Options (Options menu)

This option lets you change some runtime aspects of KeyGO.

In a normal day-to-day use of KeyGO, the Startup Options dialog box should look like this:
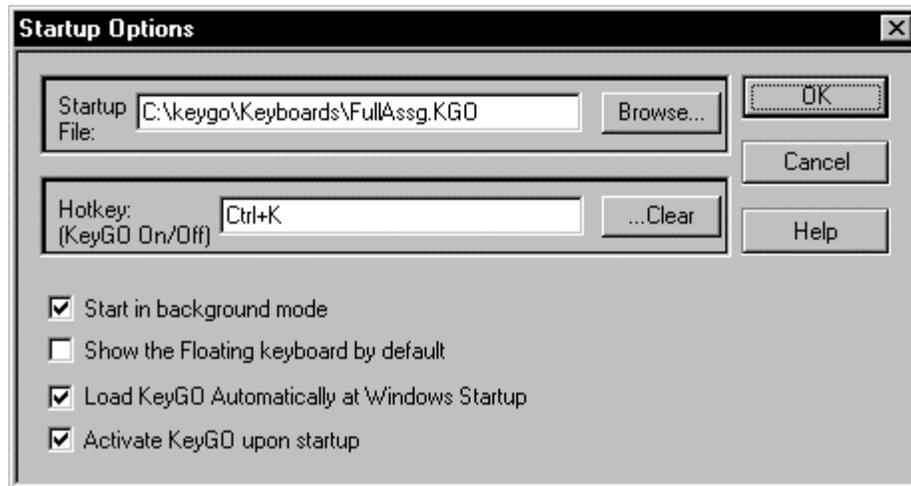
Start-up File

This dialog box lets you enter the name of the KeyGO file that will be loaded automatically each time KeyGO is run. The browse button lets you find the desired file from the disk.

Note: To be able to reactivate the keyboard layout specified in this Startup File using the hotkeys or the Floating Action Bar, this file must also be listed in the File Mapping List (see Keyboard mapping below).

Hotkey

The hotkey is a toggle key used to rapidly turn KeyGO's control of the keyboard on and off. It can be used from all windows applications. Click on this field and press any combination of keys that you want to use as the main KeyGO on/off switch key. It is highly recommended to use Ctrl, Alt and Shift key combinations. However, any key or state key combination can be used. The Clear button lets you erase the content of this box.

Start in Background mode

Use this check box if you want KeyGO to run in the background of Windows each time it is loaded. The background mode of KeyGO is the day-to-day way of using KeyGO once the proper keyboards have been build and assigned.

Show the Floating keyboard by default

If this option is set, it will force KeyGO to show the floating keyboard each time KeyGO is run, even if it was previously turned off. If this option is not set, the floating keyboard will keep its previous state from one session to the next.

Load KeyGO Automatically at Windows Startup

Use this check box to inform KeyGO that it should be loaded each time Windows 9x or NT is started.

Activate KeyGO upon startup

Use this check box to inform KeyGO to activate the keyboard layout specified in the Startup File section automatically upon startup. This box will be unavailable if the Startup File box is empty.

Keyboard mapping (Options menu)

These options let you build a list of KeyGO keyboard layout files and let you switch from one layout to another within all Windows applications by using hotkeys or the keyboard list on the Action bar or on the Windows 9x (or Windows NT) System Tray. See Keyboard assignments page 31).

Use the **Keyboard Hotkeys automatically turn KeyGO On** to inform KeyGO to activate the keyboard translation when a new keyboard is loaded from the hotkeys even if KeyGO is initially off. Note that making a selection from the Action bar list or the system tray will always turn KeyGO on.

Keyboard Summary command (Options menu)

Summary Info	✕
Filename: Cndmulti	OK
Title: Canadian Multilingual	Cancel
Comments: Windows Canadian Multilingual keyboard definition	Help
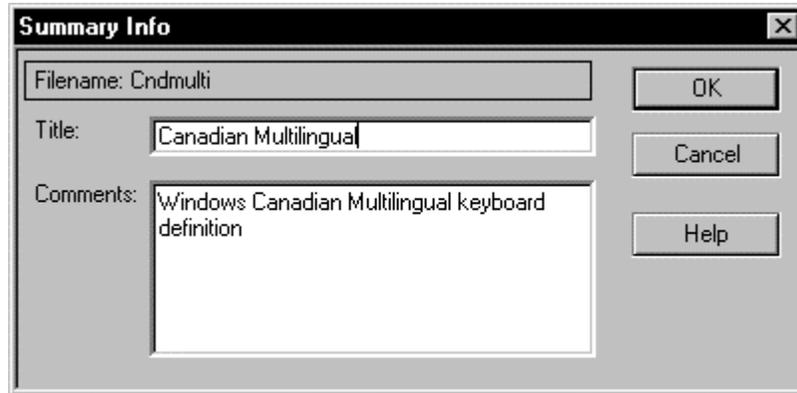
This window lets you enter keyboard file-specific information. The Title section will be used to list the keyboard in the keyboard menu activated from the **Action Bar** (see page 44) or from the Windows 9x System tray (see page 33). The **Use title, if it exists** option in the **View Preferences** menu should be set on for this to work (see View Preferences page 51).

Keyboard General Options...

Keyboard General Options ⊠

Filename: FullAssg

☑ Display selected fonts　　OK

☐ Enable Right-to-Left writing　　Cancel

ATTENTION:

Consult Help for limitations on using the Right-to-Left writing features.　　Help

Auto Activation String:

^fArial^s12^|

The **Display Selected Fonts** option, when checked, lets KeyGO show all the fonts used on the current keyboard layout. When unchecked, only a system font will be shown. Doing so, the mixing of different fonts is no longer permitted.

If your word processor does not allow you to write right-to-left, KeyGO's **Enable Right-to-Left writing** gives you this possibility. This feature is especially useful when mixing left-to-right and right-to-left writing in a single document. To properly use this feature your word processor should be set to right paragraph alignment (consult the documentation of your word processor). A space character (first character of the character map) should also be assigned to the Space Bar.

Note: Only keys which have assignments on them will be translated. The ENTER key has a special meaning. No character assigned to it will be sent to the application if the Right-to-left feature is enabled. To validate the ENTER key as a Right-to-left ENTER key in a non right-to-left writing application, assign any character to it.

Example: To combine two languages in a document, build two keyboard layouts, one for each language with the proper assignments on the keys. The space bar and the ENTER key on the right-to-left writing language keyboard should have assignments on them. Add a value in the **Auto Activation String** box corresponding to the sequences of keys necessary to switch the word processor to the proper font for each keyboard (ex: ^+fSymbol~ under Winword 6.0 will switch the font to Symbol each time the keyboard is called using their associate hortkey) . Enable the Right-to-Left option for the proper keyboard layout. Save the two files and add them with hotkeys to the File list in the Keyboard Mapping Options dialog box of the Options menu. From now on, you can use both language keyboard layouts in your document and switch between both languages using the hotkeys (or the FileList of the Action Bar). The fonts will be automatically selected with the proper support for the right-to-left writing.

The **Auto Activation String** field lets you enter a string (as explained in the chapter on strings page 19) that will be sent to the application each time the keyboard layout is activated through the use of hotkeys. Note that the Auto Activation string will not be sent to the application if the keyboard is loaded using the File List of the Action Bar on the System Tray.

This feature is very useful to automatically select a font in your application when a new keyboard template is selected.

Example: The string ^+fsymbol{enter} would shift Winword 6.0 to the Symbol font each time the keyboard associated with it is loaded using a hotkey (see the chapter on strings page 19 for details on the different macro commands)

Keyboard Language options (Options menu)
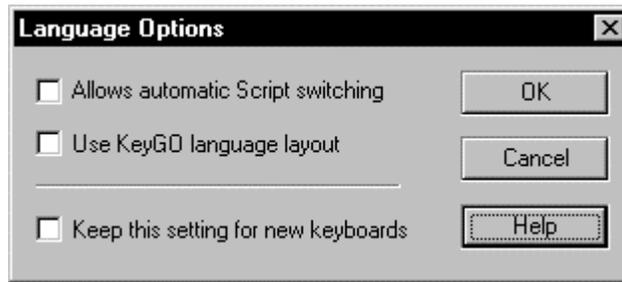
This Option dialog box lets you choose options concerning the language support of KeyGO.

KeyGO 2.x will automatically switch between script/fonts if your application has multilanguage support (Word 7.0, Wordpad. Etc.). For this feature to work you have to make sure that the **Automatic script switching** option is enabled (See page 31)

If the option **Use KeyGO language layout** is set in the Keyboard Language Options dialog box, every time an <u>un</u>assigned key is depressed, a language switching command corresponding to the one used to lay out the keyboard template will be sent to the application. In essence, KeyGO will force your application to switch to the language specified by the layout of your keyboard, replacing all Windows 9x defaults. The keyboard language can be chosen from the **Language** pull-down menu from the **Tools** menu (see page 46).

The **Keep this setting for new keyboards** option, when validated, will make the current selection the default one for all new constructed keyboards.

View Preferences options (Options menu)

This dialog box will pop up when the View preferences dialog option is selected from the Options pull down menu. Not all options in this menu will be available when using version 3.5x of Windows NT.

The **Big Buttons** option when validated will increase the size of the toolbar used in KeyGO builder.

The **Scale-to-fit by default** option when validated will force the keyboard layout to match the size of its surrounding window each time a keyboard is opened or a new keyboard layout is created.

The **Hide KeyGO in background** option lets you choose the appearance of KeyGO when it is put in the background. If hidden, KeyGO will not be seen on the Windows 9x Taskbar. To be sure that there is still a possible access to KeyGO, the Show KeyGO access icon will be automatically turned on, and the access icon will show up on the Windows 9x System tray. To make KeyGO completely hidden, unmark the Show KeyGO access icon option.

At this point, the only way to call KeyGO's builder when KeyGO is in the background would be to call KeyGO from the Windows 9x Start button.

The **Use floating Action bar** and the **Use Windows 9x Taskbar** options let you choose between using the floating Action or the Windows 9x Taskbar to show KeyGO access icons.

The **Use title, if it exists, to list keyboard in menu** option will show the keyboards listed in the menu by their titles as entered in the Keyboard summary option dialog box. The keyboard menu is accessed from the icon, either on the Taskbar or on the Floating Action bar.

The **Show KeyGO On/Off icon** will show or hide the icon on the Taskbar that lets you turn KeyGO on and off.

The **Show Keyboard show/hide** icon will show or hide the icon on the Taskbar that controls the visibility of the floating keyboard.

The **Show Keyboard** menu icon will show or hide the icon on the Taskbar that brings up the keyboard list menu.

The **Show KeyGO access icon** will show or hide the icon on the Taskbar that calls up or puts KeyGO to the background.

OLE Transfer Options (Options menu)

This options dialog box lets you choose which method will be used by KeyGO to send an OLE object to an application. If the **Activate OLE Object** option is checked, all new inserted objects assigned to keys will be activated instead of pasted to the application. However, if the **Copy and Paste** method is checked, all new objects will be sent to the application through the copy and paste Windows' process.

To change the way a specific assigned object should behave, double-click on the keycap containing the object (or click the right mouse button) to get the OLE Activation dialog box.

The **OLE Copy and Paste Method** option lets you choose which command will be used to paste an OLE object to the application. Most applications will respond well to the Windows' Paste command. As for other applications, different methods have to be tried. For instance, Winword 7.0 will not respond to a Paste command, but will have no problems responding to the Ctrl-V or Shift-Ins command. For more details on OLE assignments, see page 23)

Character Transfer Options (Options menu)
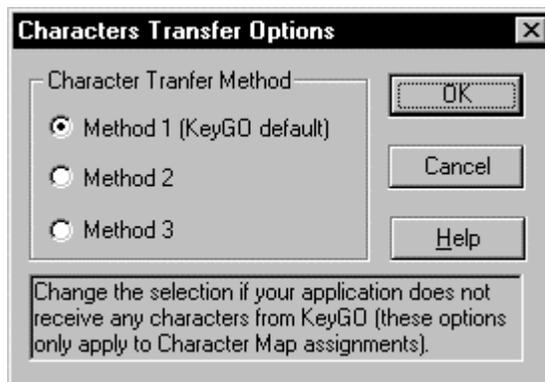
This options dialog box lets you choose which method will be used by KeyGO to communicate character assignments to Windows' applications. Method 1 will normally be proper for all Windows' applications. However, there are some rare cases where KeyGO does not yield the proper character to the application. If this happens, try to use KeyGO with methods 2 and 3.

Note that these options affect only assignments done from the character map. They do not affect the OLE or the string assignment behavior.

New Windows (Windows menu)

You can create a new view window of the same keyboard layout by using this command.

Cascade command (Window menu)

Use this command to arrange multiple opened windows in an overlapped fashion.

Tile command (Window menu)

Use this command to arrange multiple opened windows side by side.

Arrange Icons command (Window menu)

Use this command to arrange the icons for Minimized windows at the bottom of the main window. If there is an open keyboard layout window at the bottom of the main window, then some or all of the icons may not be visible because they will be underneath this keyboard layout window.

1, 2, ... command (Window menu)

KeyGO displays a list of currently opened keyboard layout windows at the bottom of the Window menu. A check mark appears in front of the keyboard layout name of the active window. Choose a keyboard layout from this list to make its window active.

Help Main Index command (Help menu)

Use this command to display the opening screen of Help. From the opening screen, you can access step-by-step instructions for using KeyGO as well as various types of reference information.

Once you open Help, you can click the Contents button whenever you want to return to the master index of the KeyGO help.

Using Help command (Help menu)

Use this command for instructions about using Help.

About command (Help menu)

Use this command to display the copyright notice and version number of your copy of KeyGO.

Appendix A

KEYGO registry Specifications

These entries in Windows registries are done under:

HKEY_CURRENT_USER\Software/GDGSYSTEMS/KeyGO

You can use the program "regedit" (Start, run) to modify these values. However, it is not recommended to change them manually.

[Settings]	
ShowToolbar	0 - Hide the toolbar
	1 - Show the toolbar
ShowStatusBar	0 - Hide the status bar
	1 - Show the status bar
ShowDeadkeysList	0 - Hide the Deadkey list box
	1- Show the Deadkey list box
ShowCharMap	0 - Hide character map
	1 - Show character map
ShowStrings	0 - Hide Custom string box
	1 - Show Custom string box
WindowPos	Different keyboard layout window size and positions
ShowCmd	0 - Show KeyGO minimized when run(=the last state of KeyGo before closing).
	1 - Show KeyGO maximized when run.
ShowActionWindow	0 - Hide the floating Action bar
	1 - Show the floating Action Bar
StringPosX, StringPosY	Coordinate of the String box.
CharmapPosX, CharmapPosY	Coordinate of the Character map
DeadListPosX, DeadListPosY	Coordinate of the Deadkey list box
ActionPosX, ActionPosY	Coordinate of the Action bar
LastViewSizeX, LastViewSizeY	Size of the builder Window
LastViewPosX, LastViewPosY	Position of KeyGO
DeadListSizeX, DeadListSizeX	Size of the deadkey list box
LastViewType	KeyGO view type (minimized, maximized, etc.)
PreviewPages	The number of pages used during the last print preview
[OLE]	
OLELaunchMethod	Define the OLE launching method (0- Activate OLE, 1- Paste OLE)
OLECopyPasteMethod	Define the way KeyGO will paste the OLE object (0,1 or 2)
[Languages]	Language related options
SwitchFont	0 - Do not switch the script/font of the target API
	1 - Do it
Use KeyGOKbdLayout	0 - Do not use the default language of the active keyboard
	1 - Use it
[Small Keyboard]	
X	Last X position of the floating keyboard
Y	Last Y position of the floating keyboard
[Hotkey]	
Scancode	Scancode for the activate Hotkey

Shiftstate	Shift key for KeyGO Hotkey
[Mappings]	Information on the keyboard assignments
MapCount	The number on entries in the map
ScanCode00	Scan code of the Hotkey for the first file in the map
ShiftState00	Shift key for the Hotkey of the first file in the map
FileName00	Name of the first file in the map
ScanCode01	Scan code of the Hotkey for the second file in the map
ShiftState01	Shift key for the Hotkey of the second file in the map
FileName01	Name of the second file in the map
[General]	From the options submenu
KeyGOAutoOn	0 - Do not turn KeyGO On at load time 1- Turn KeyGO On at load time
AllwaysScaled	0 -Do not automatically scale-to-fit the keyboard 1 - Scale-to-fit new opened or created keyboard
ActionBarOn	0 - Action Bar is Off 1 - Action Bar is On
DefaultKeyboardType=?	The default keyboard type when creating a new keyboard
UseActionBar	0 - Use the System tray for KeyGO access icons 1 - Use the Action bar for KeyGO access icons
KeyboardOn	0 - Keep last view state of the floating keyboard 1- Turn the floating keyboard On by default at startup
AutoLoad	0 - Do not load KeyGO at Windows startup 1 - Load KeyGO automatically at Windows startup (This option should only be changed from the startup menu)
AutoActivate	0 - Do not activate KeyGO at load time 1- Activate KeyGO at load time
UseMarker	0 - Do not use markers on the characters map 1 - Use the marker in the character map
BigButtons	0 - Use small icons on toolbar 1- Use big icons on toolbar
CharMethod	0,1,2 - The three different method of character transfer
StartupFile	File to load and use at KeyGO load time
RunMinimized	0 - Don't run minimized (from the Option box) 1 - Run minimized
ShowSmallWindow	0 - Hide floating keyboard 1 - Show floating keyboard
FrcJrnPlay	1 - Force KeyGO to use the Journal Playback of Windows to send all characters assignments (an alternative when having some problems). Unused in Version 2.x.
[MenuBar]	View preference for keyboard list and system tray
KeyGOOnOff	0 - Do not show KeyGO On/Off icon 1- Show it
FileList	0 - Do not show the keyboard list icon 1 - Show it
SmallKeyboard	0 - Do not show the floating keyboard On/Off icon 1 - Show it
HideKeyGO	0 - Do not hide KeyGO when minimized (in background) 1 - Hide it
KeyGOCall	0 - Do not show the KeyGO builder access icon 1 - Show it
MenuWithTitle	0 - Do not show keyboards in list with their title 1 - Show keyboards in list using their title

[Recent File List]	The last four files used

Appendix B

Windows commands

Size command (System menu)

Use this command to display a four-headed arrow so you can size the active window with the arrow keys. ⊕

After the pointer changes to the four-headed arrow:

1. Press one of the DIRECTION keys (left, right, up, or down arrow key) to move the pointer to the border you want to move.

2. Press a DIRECTION key to move the border.

3. Press ENTER when the window is the size you want.

Note: This command is unavailable if you maximize the window.

Shortcut

Mouse: Drag the size bars at the corners or edges of the window.

Move command (Control menu)

Use this command to display a four-headed arrow so you can move the active window or dialog box with the arrow keys. ⊕

Note: This command is unavailable if you maximize the window.

Shortcut

Keys: CTRL+F7

Minimize command (application Control menu)

Use this command to reduce the KeyGO window to an icon.

Shortcut

Mouse: Click the Minimize icon ▬ on the title bar.

Keys: ALT+F9

Maximize command (System menu)

Use this command to enlarge the active window to fill the available space.

Shortcut

Mouse: Click the Maximize icon on the title bar; or double-click the title bar.

Keys: CTRL+F10 enlarges a keyboard layout window.

Next Window command (keyboard layout Control menu)

Use this command to switch to the next open keyboard layout window. KeyGO determines which window is next according to the order in which you opened the windows.

Shortcut

Keys: CTRL+F6

Previous Window command (keyboard layout Control menu)

Use this command to switch to the previous open keyboard layout window. KeyGO determines which window is previous according to the order in which you opened the windows.

Shortcut

Keys: SHIFT+CTRL+F6

Close command (Control menus)

Use this command to close the active window or dialog box.

Double-clicking a Control-menu box is the same as choosing the Close command.

Note: If you have multiple windows open for a single keyboard layout, the Close command on the keyboard layout Control menu closes only one window at a time. You can close all windows at once with the Close command on the File menu.

Shortcuts

Keys: CTRL+F4 closes a keyboard layout window

ALT+F4 closes the Keyboard window or dialog box

Restore command (Control menu)

Use this command to return the active window to its size and position before you chose the Maximize or Minimize command.

Appendix C

Troubleshooting

PROBLEM	ANSWER
KeyGO does not work.	Be sure that KeyGO is activated with the proper keyboard layout by clicking ⬤ on the toolbar. The name of the active keyboard should appear on the status bar at the bottom of the KeyGO window.
KeyGO does not work in some specific applications.	Some Windows applications install their own keyboard drivers. To force the characters sent by KeyGO to these applications try different character transfer methods from the Character transfer option dialog box in the option menu.
I get a "General Protection Fault" when using KeyGO with the Natural Keyboard from Microsoft.	Some Windows drivers are not supported by the Natural Keyboard and may give erratic results when used with KeyGO. Choose a keyboard driver that is well supported by the Natural Keyboard in the IntelliType program (example: US keyboard).
The captions on the keycaps of KeyGO are not the same as my physical keyboard.	If you are using the Natural Keyboard from Microsoft, see above. If not, reinstall the keyboard driver of Windows or choose another keyboard layout from the keyboard type dialog box shown when creating a new keyboard layout.
One or some of my assignments do not work.	You must make sure that there are NO hotkeys assigned to the keys on which the assignments are done. Go to the Option dialog box of the Keyboard menu to consult the different hotkey assignments. And check that Windows 9x is not already using these keys as hotkeys.
It is not possible to load a keyboard template using its corresponding hotkeys (as entered in the option dialog box).	You must make sure that there are NO other assignments under this key combination on the keyboard layout and that Windows 9x is not using these keys as hotkeys.
The characters are not sent with the proper font to the application.	You must manually choose the font in your application.
A sequence of control characters is sent to the application instead of the command entered in the String dialog box.	There is an error in your control sequence. Check it out.
It is not possible to get the proper assignment language in my application	Make sure that your application supports the multi-language specifications of Windows.
	Make sure that the options in the Language options dialog box are set properly.
	When assigning a Western type of character to a key, the

	language reference that will be sent to the application will be the first found in the list of keyboards (in the Keyboard options of the control panel) that matches the Western script. If you want this to be different, choose a new Western keyboard template in the keyboards options of Windows as the default one.
My application is beeping when using KeyGO	Make sure that the Character transfer option in the Options menu is set to Method 1. A String will always beep when sent to Winword 7.0. There is no fix for this.
The application does not understand the control sequence sent to it.	Some Windows applications do not understand keystroke sequences sent to them at full speed. Insert a 1 millisecond delay ([1]) before the control code that has a problem (see page 19).

Index