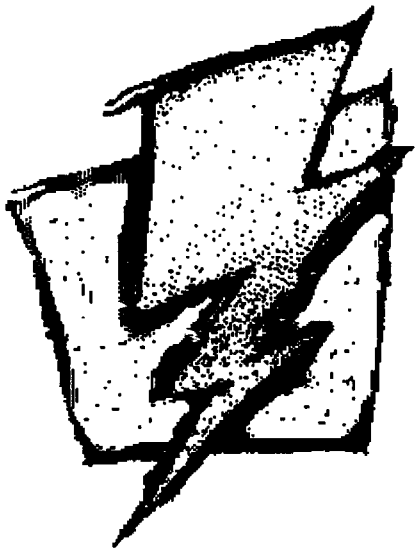
WATCOM VX REXX FOR OS/2

ERIC GIGUERE
Waterloo



WATCOM VX REXX

The Complete REXX Programming Environment for OS/2

Eric Giguère
WATCOM International
giguere@csg.uwaterloo.ca

The REXX language provides no facilities for user interaction other than through a simple console to which a program can read and write lines of text. On an advanced system like OS/2, these facilities seem primitive in comparison to those offered by the Presentation Manager (PM). VX REXX breaches this gap to let you develop REXX-based PM applications.

Features

WATCOM VX REXX is a complete REXX development environment with the following features:

- **Project management facilities.** A *project* is the REXX code and the Presentation Manager user interface that together form an application. VX REXX allows you to work on the project in small, individual pieces without limiting your ability to view and edit the complete REXX program.

- **Direct user interface design and editing.** VX REXX lets you directly design your program's user interface using *objects* based on standard Presentation Manager windows and controls. You create, size and position the objects and modify their settings as you would with the Workplace Shell objects on your desktop.

- **Testing and debugging.** Test your application directly within the VX REXX environment. If an error occurs, VX REXX will show you exactly where it happened. Use the VX REXX symbolic debugger to track errors in program logic.

- **Useful REXX extensions.** VX REXX includes sets of functions for displaying standard dialogs, performing common file operations, and creating and manipulating objects.

- **Flexibility and extendability.** VX REXX can generate standalone executables or macros for use with other applications. Your programs can use third-party REXX function packages and other REXX extensions that follow the application programming interface (API) defined by OS/2. New object types can also be added to VX REXX in C using SOM.

WATCOM VX REXX works with the standard OS/2 2.0 REXX interpreter.

A Simple Example

Figure 1 illustrates the VX REXX development environment. A few simple steps are all you need to write your first VX REXX application:

1. Select the PushButton item from the Tools menu or click on the appropriate icon in the tool palette. Move over the gridded window and press the left mouse button to position and size a new PushButton object.

2. Press the right mouse button on the PushButton object to bring up its popup menu. Choose the Properties item to bring up the PushButton's properties notebook.

3. Set the Caption property to the string "Push Me!" by selecting the Text tab of the notebook and typing the string into the Caption entry field. (See Figure 2.)

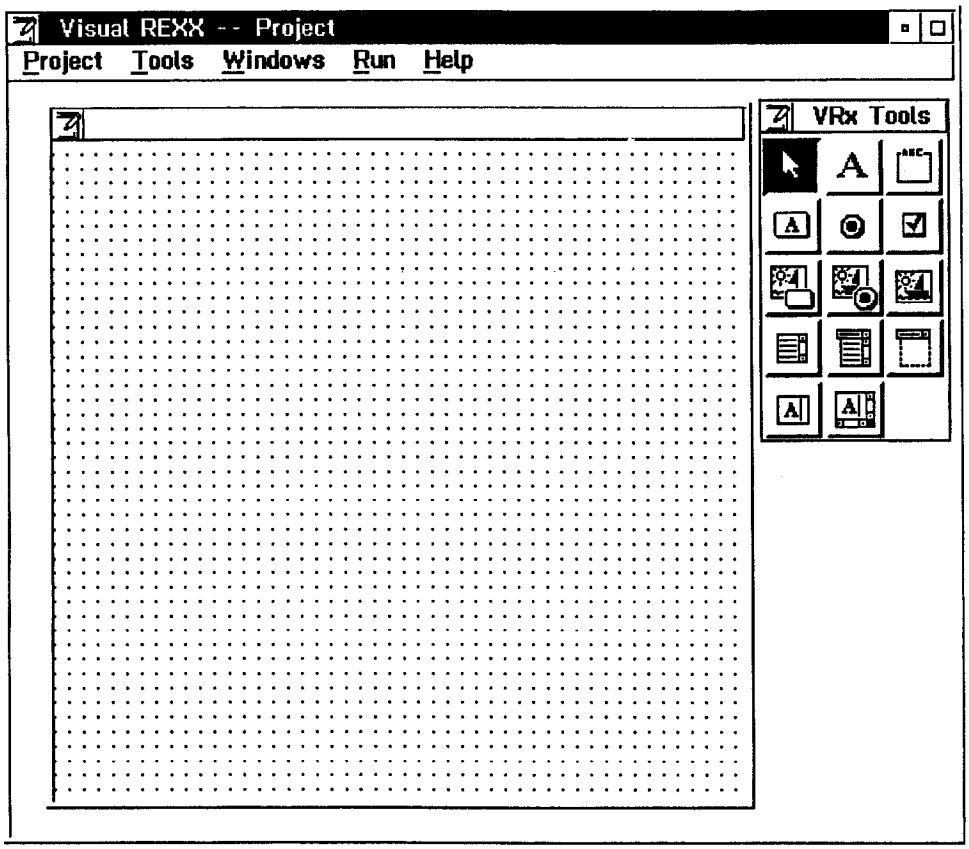


Figure 1: The VX REXX development environment.

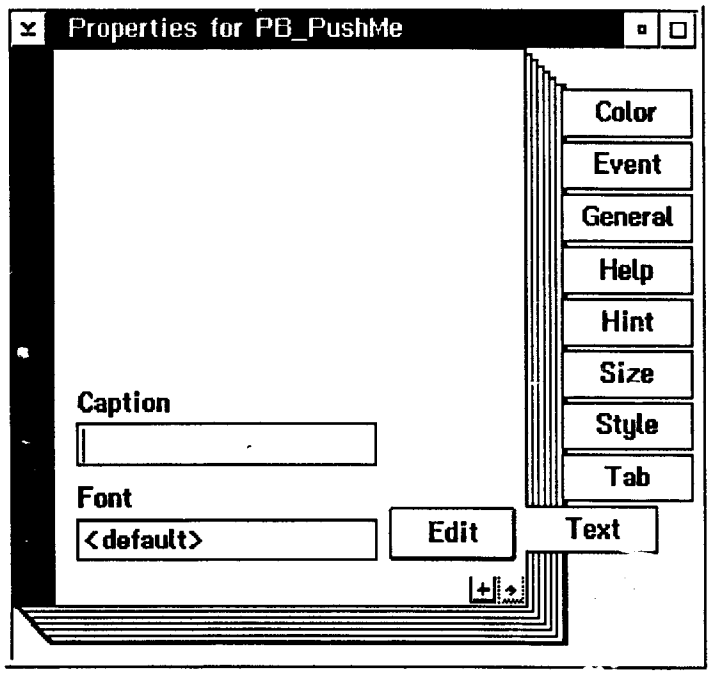


Figure 2: Changing the caption of a push button object.

4. Select the Event tab and double click on the Click item in the listbox. A text editor will appear with the skeleton for a REXX procedure. Type `say "You pushed me!"` after the procedure label and close the text editor window.

Your application is now complete. To test it, select the Run Project item from the Run menu. Press the push button several times. VX REXX will automatically open a console window when the first `say` insruction is executed. When you are finished, simply close the window to halt the REXX program.

When you are satisfied with your application, select the Make EXE item from the Project menu to build a standalone executable.

For More Information

For more information on VX REXX, contact WATCOM at:

WATCOM
415 Phillip Street
Waterloo, Ontario
CANADA N2L 3X2

Phone: (519) 886-3700
Fax: (519) 747-4971
BBS: (519) 884-2103