

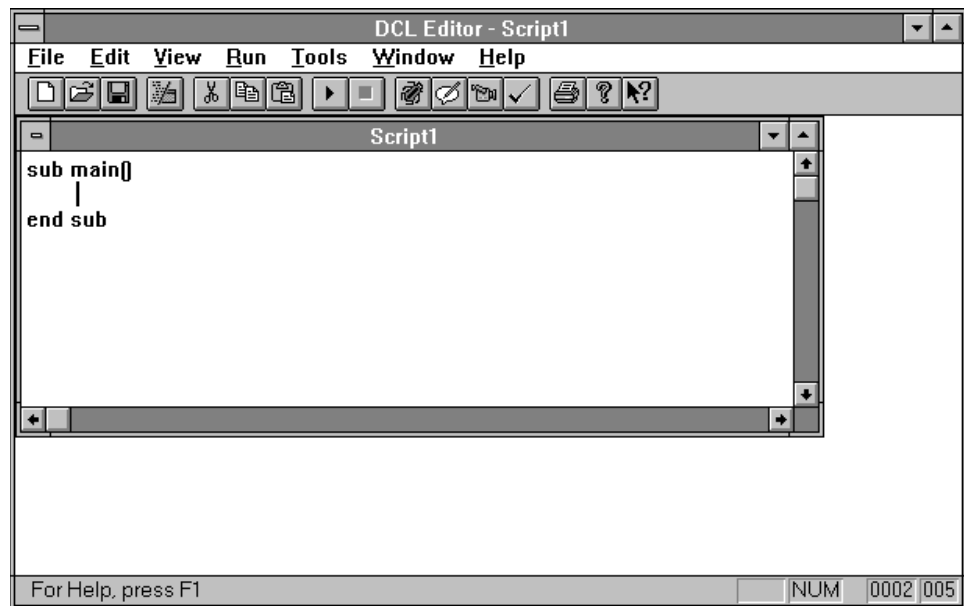
Using the DCL Editor

DCL provides a multiple document interface (MDI) to ease the process of editing, testing, and debugging scripts. This chapter describes how to use the **DCL Editor** to create, modify, and test your scripts.

Starting the DCL Editor

To start the **DCL Editor** (DCLEEDIT.EXE), choose the DCL Editor icon from the Applications Manager window (or Program Manager window).

The **DCL Editor** window is displayed.



The DCL Editor window

Exiting DCL Editor

To exit **DCL Editor**, choose the Exit command from the File menu.

Creating a New Script

Use the New command to open a window to create a new script.



To create a new script

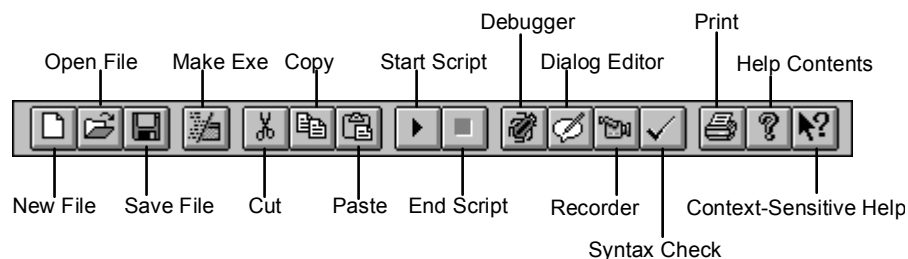
1. Choose the New command from the File menu.
2. Enter the commands that make up the script.
If you need help for a particular command, position the cursor on the line containing the command and press F1.
3. Save your script as described under “Saving a New Script” in this chapter.
4. Run your script as described under “Running a Script” in this chapter.

Using the Toolbar and Status Bar

DCL provides a Toolbar and Status Bar for your convenience in developing scripts.

Using the Toolbar

The Toolbar is a row of buttons representing frequently used **DCL Editor** commands. Instead of using the menus, you can choose a Toolbar button to execute a command.



The DCL Editor Toolbar

To toggle the Toolbar display on or off, choose Toolbar from the View menu.

Using the Status Bar

The Status Bar appears at the bottom of the **DCL Editor** window.

When a menu is displayed, the Status Bar provides a description of the currently selected item.

The Status Bar also indicates when the Caps Lock or Num Lock key is on. The current line and column of the cursor are displayed in far right corner of the Status Bar.

To toggle the display of the Status Bar on or off, choose Status Bar from the View menu.



The DCL Status Bar

Working With Files

The scripts you write are saved as ASCII (text) files.

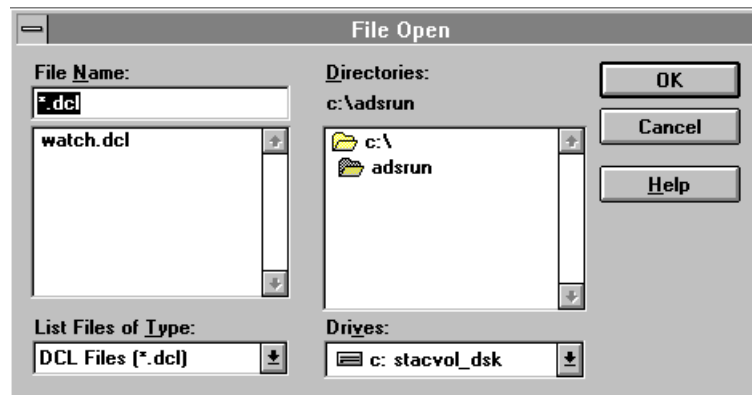
Opening an Existing File

Use the Open command to open an existing file.



To open an existing file

1. Choose the Open command from the File menu.



2. Select the Drive and Directory containing the file.
3. Enter the File Name. Or select the file from the File Name list box.

You can change the types of files displayed in the File Name list box by selecting another file type from the List Files of Type box.

4. Choose OK.

Opening a Recent File

The last four files you saved are displayed toward the bottom of the File menu.

To open one of these files, choose the file name from the File menu.

Closing a File

Use the Close command to close the current script. If you have made changes to the file, you will be asked whether you want to save the changes.

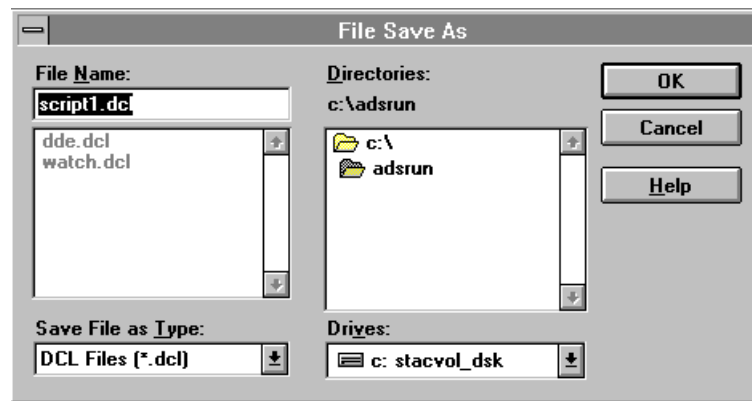
Saving a New Script

Use the Save command to save the current new script.



To save a new script

1. Choose the Save command from the File menu.



2. If you need to change the current path, select the Drive and Directory to contain the file.
3. Enter the File Name.
4. Choose OK.

Saving an Existing File

To save an existing file, choose the Save command from the File menu.

Saving an Existing File Under a New Name

Use the Save As command to save an existing script under a new file name.



To save an existing file under a new name

1. Choose the Save As command from the File menu.
The File Save As dialog box is displayed.
2. If you want to change the current path, select the Drive and Directory to contain the file.
3. Enter the File Name.
4. From the Save File As Type box, select the desired file type.
5. Choose OK.

If a file having the same name already exists, you will be asked to confirm that you want to overwrite the file.

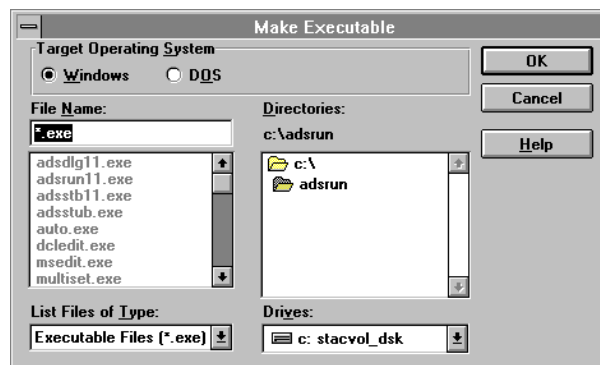
Making an Executable

Use the Make Exe command to save the current script as an executable file.



To make an executable from the current script

1. Choose the Make Exe command from the File menu.



2. Select the Target Operating System – Windows or DOS.
3. If you want to change the current path, select the Drive and Directory to contain the file.

4. Enter a new File Name or accept the default executable name.
5. Choose OK.

If a file having the same name already exists, you will be asked to confirm that you want to overwrite the file.

Note: If you create an executable to run under DOS, make sure the **DCL** commands in your script are supported under DOS. Chapter 8, “Command Reference,” indicates if a command is **not** supported under DOS.

Distributing an Executable

When you distribute an executable to users in your organization to run under Windows, the following files must be in the same directory as the executable or on the path:

- ADSCON11.EBL
- ADSENV.DLL
- ADSPUB11.DLL
- ADSRUN11.DLL
- ADSUTILS.DLL
- CTL3D.DLL
- DCL.EBL
- WUNETWAR.DLL (For network functionality)
- WUNET.INI "
- NWCALLS.DLL "
- NWLOCALE.DLL "
- NUNET.DLL "

No support files are required for executables compiled to run under DOS.

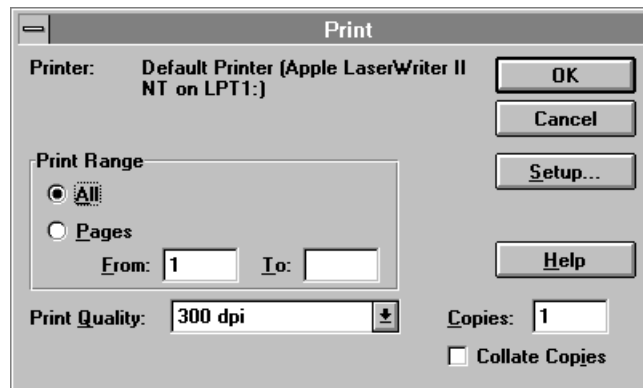
Printing a Script

Use the Print command to print all or part of the current script.



To print a script

1. Choose the Print command from the File menu.



2. If you want to change the printer or printer settings, choose the Setup button and see “Setting Up for Printing.”
3. Select the Print Range. Your choices are all of the file or a range of pages.
4. Change any of these print options as desired: Print Quality (depending on your printer), Copies, Collate Copies.
5. Choose OK.

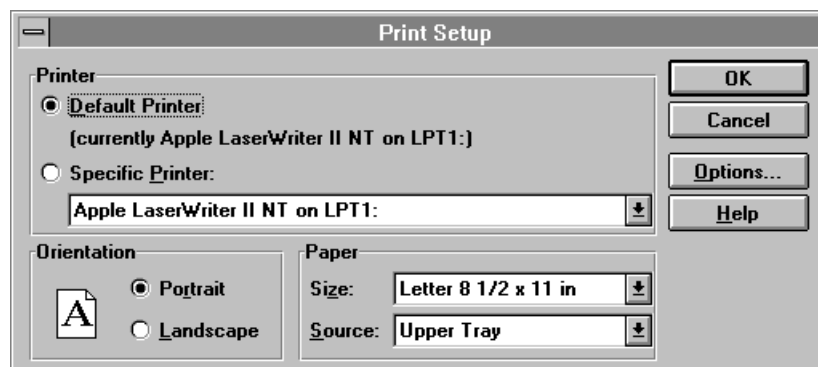
Setting Up for Printing

Use the Print Setup command to set up your printer.



To set up your printer

1. Choose the Print Setup command from the File menu.



2. Select the Default Printer or another Specific Printer.
3. Select the Orientation of the paper—portrait or landscape.
4. Select the desired Paper Size and Source, depending on your printer.
5. If you want to change the setting for your printer, choose the Options button. One or more dialog boxes specific to your printer will be displayed.
6. Choose OK.

Editing Text

The **DCL Editor** functions like other Windows text editors.

Use the Cut command on the Edit menu to remove a block of text you have selected from your script and store it in the Windows clipboard.

Use the Copy command to copy the selected text to the clipboard.

Use the Paste command to insert the text currently in the clipboard at the insertion point in the current document window.

Use the Select All command to select all of the text in the current document window.

Use the Move Text Left and Move Text Right commands to shift the selected text one tab stop left or right.

Use the Undo command to undo the last editing operation.



To select a block of text for an editing operation

1. Position the insertion point at the beginning of the text to be selected and click the left mouse button.
2. Move the insertion point to the end of the block to be selected.
3. Hold down the Shift key and press the left mouse button.

OR

1. Move the insertion point to the beginning of the text to be selected.
2. Hold the Shift key.
3. Move the cursor to the end of the text to be selected.

You can choose the Select All command from the Edit menu to select all of the text in the current document window.

Finding Text

Use the Find command to search for a text string in the current script.



To find a string of text in a script

1. Choose the Find command from the Edit menu.



2. Enter the text string to search for in the Find What text box.

Shortcut: If you select the text you want to search for before choosing the Find command, the selected text is put in the Find What text box for you.

3. Select the Match Case check box if you want the search to differentiate between upper- and lowercase letters.
4. Specify the Direction of the search—up or down.
5. Choose the Find Next button to begin the search.

The first occurrence of the string after the insertion point is highlighted.

6. To continue searching for subsequent occurrences of the string, choose Find Next.

The next occurrence of the string is highlighted.

Hint: You can edit the script the with the Find dialog box displayed. Use the mouse or press ALT+F6 to move between the dialog box and the script.

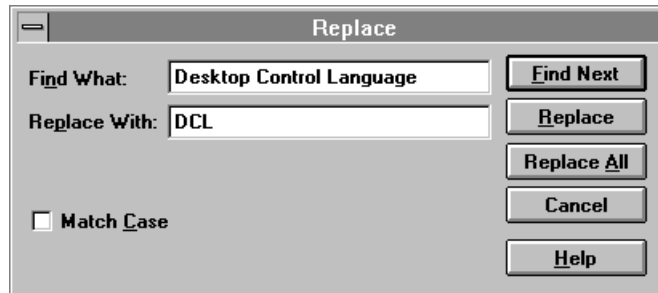
Replacing Text

Use the Replace command to replace one text string with another in the current script.



To replace one text string with another

1. Choose the Replace command from the Edit menu.



2. Enter the text string to search for in the Find What text box.

Shortcut: If you select the text you want to search for before choosing the Replace command, the selected text is put in the Find What text box for you.

3. Enter the replacement text in the Replace With box.
4. Select the Match Case check box if you want the search-and-replace operation to differentiate between upper- and lowercase letters.
5. Choose the Find Next button to begin the search.

The first occurrence of the string after the insertion point is highlighted.

6. If you want to substitute the replacement text for the string, choose the Replace button. (The search for the next matching string continues after the replacement is made.)

OR

If you want to go on to the next occurrence of the string without making the text change, choose the Find Next button.

OR

If you want to change all subsequent occurrences of the string to the replacement text, choose the Replace All button. (To avoid accidental changes, we recommend that you use this feature with caution.)

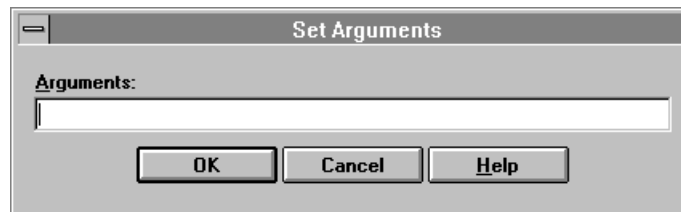
Running a Script

To run a script that does not use command-line arguments, choose Start Script from the Run menu.



To run a script that uses command-line arguments

1. Choose the Arguments command from the Run menu.



2. In the Set Arguments dialog box, enter the command-line arguments to be passed to the script's main subroutine and choose OK. (In your script, you can get these arguments through the `Command$` statement, which is described in Chapter 8, "Command Reference.")
3. Choose Start Script from the Run menu.

To stop the script that is currently executing, choose End Script from the Run menu.

Running Executables

If the script has been saved as an executable file (using the File/Make Exe command), you can use the **Applications Manager** or Program Manager File/Run function or File Manager to run the script. For more information, see "Making an Executable" in this chapter.

Using DCL Tools

DCL provides a Debugger, Dialog Editor, Syntax Checker, and Macro Recorder.

Starting the Debugger

To start the **DCL Debugger**, choose Debugger from the Tools menu. For details on how to use the Debugger, see Chapter 5 "Using the Debugger."

Starting the Dialog Editor

To start the **DCL Dialog Editor**, choose Dialog Editor from the Tools menu. For details on how to use the Dialog Editor, see Chapter 4, "Using the Dialog Editor."

Checking the Syntax

You can check the syntax of your script to remove any errors.



To check the syntax of your script

1. Position the insertion point (cursor) at the beginning of the script.
2. Choose Syntax Check from the Tools menu.

Upon encountering an error, the syntax checker stops at the line containing the error and displays an error message in a message box and in the Status Bar.

3. Correct the error.
4. Repeat Steps 1 – 3 until your script is free of syntax errors.

If your script is free of errors, the message “Syntax OK” is displayed in the Status Bar.

Recording a Macro

DCL’s Recorder captures Windows events and translates them into **DCL** statements that can be inserted in a script. For a list of the of the events you can record, see “Recorded Events” in this chapter.



To record Windows events

1. Choose Recorder from the Tools menu.



2. Make any desired changes to the dialog box:

Include Comments: Indicate whether to include descriptive comments in the code generated by the Recorder.

High-Level Statements: Indicate whether to consolidate multiple events into one High-Level Statement where possible. (For example, selecting a menu is recorded as a menu command rather than a series of mouse commands.)

Keyboard: Indicate whether to capture events resulting from keyboard activity.

Mouse: Indicate whether to capture events resulting from mouse activity then indicate whether the coordinate system is relative to the window or screen.

Stop Recording On: Specify the key combination to stop the Recorder.

3. Choose OK.

The **DCL Editor** window is minimized. The Recorder window is displayed.



4. Perform the Windows events you want to record.

You can interrupt recording by choosing the Pause button in the Recorder window. Then choose the Record button when you want to resume recording.

5. When you have finished recording events, choose the End button in the Recorder window.
6. The **DCL Editor** window is restored; the Insert Recording dialog box is displayed.



7. Press F9 (or use mouse) to switch to your script.
8. Position the insertion point where you want to insert the recorded code.
9. Press F9 (or use mouse) to switch back to the Insert Recording dialog box.
10. Choose OK.

Recorded Events

The Recorder recognizes the Windows actions described in the following paragraphs. For information about **DCL** commands, see Chapter 8, “Command Reference.”

Application Focus Switch

The Recorder recognizes when an application receives the focus and records this as a `WinActivate` statement if you perform an activity within the application. The focus can be shifted using any of the normal Windows application switching methods (ALT+TAB, CTRL+ESC, clicking the mouse on another application, selecting the “Switch To...” command from an application’s control menu, etc.).

Window focus shifts are also recorded to ensure that subsequent actions occur (and are synchronized) within the appropriate window.

Statements

- `WinActivate`
- `AppActivate`

Window Scrolling

Interactions with windows that respond to scrolling messages (windows that have built-in Windows scroll bars, like Notepad and Write) are recorded.

Statements

- `VLine`
- `VPage`
- `HLine`
- `HPage`
- `HScroll`
- `VScroll`

Mouse Activity

Mouse down and up actions are compressed to high-level forms when possible. Further, mouse activity that leads to a recognizable result, such as resizing a window, is compressed out completely.

Statements

- `QueFlush`
- `QueMouseClicked`
- `QueMouseDown`

- `QueMouseDblDn`
- `QueMouseDn`
- `QueMouseMove`
- `QueMouseUp`
- `QueSetRelativeWindow`

Keyboard Activity

Keyboard press/release actions are converted. Complex key combinations, such as CTRL+ESC or ALT+SHIFT+C are converted into a readable form. Keyboard activity that results in a higher level command are compressed out completely, such as the following keystroke sequence, which results in a window reposition: ALT+Spacebar, M, Right, Right, Right, Enter.

Keyboard activity that modifies mouse events is also recorded, such as holding down the Shift key and clicking the left mouse button.

Statements

- `DoKeys`
- `QueFlush`
- `QueKeys`
- `QueKeyDn`
- `QueKeyUp`

Window Management

The Recorder recognizes high-level interactions with windows, such as movement, sizing, activation, minimizing, maximizing, and restoring. The Recorder differentiates between interactions with applications, popup windows, and child MDI windows.

Statements

- `AppMaximize`
- `AppMinimize`
- `AppRestore`
- `AppMove`
- `AppSize`
- `WinMove`
- `WinSize`
- `WinMaximize`

- WinMinimize
- WinRestore

Menu Commands

The Recorder watches for interactions with an application's built-in menu and records the results of these interactions. All intermediate events used to select the menu choice (with the keyboard and mouse) are not recorded.

Statement

- Menu

Dialog Box Interaction

The Recorder recognizes interactions with standard Windows controls, such as edit boxes, check boxes, options buttons, list boxes, and combo boxes. Also, interactions with known controls of other applications are supported, such as Borland's custom controls and the controls in Visual Basic.

Note: The results of dialog box interactions are recorded rather than the mouse/keyboard actions which resulted in those changes.

The Recorder also recognizes mixed dialogs—dialog boxes with a mixture of standard controls and custom controls. Many applications such as Control Panel, Corel Draw, and Lotus Ami Pro, use this technique. In this case, standard control interactions are recorded normally, while interactions with custom controls are recorded using high-level mouse and keyboard statements.

Statements

- ActivateControl
- SelectComboBoxItem
- SelectButton
- SelectListboxItem
- SetCheckbox
- SetEditText
- SetOption

Working with Document Windows

DCL Editor allows you to have several scripts open for editing at the same time. Each script is contained in a document window. Document windows can be moved, resized, maximized to fill the entire screen, minimized to an icon, or restored.

The names of the currently open scripts are displayed at the bottom of the Window menu. To switch to another open script, choose the script name from the Window menu or click inside the window containing the script.

To display the document windows in a stair-step arrangement, choose the Cascade command from the Window menu.

To display the document windows in a tiled arrangement, choose Tile from the Window menu.

To neatly arrange the icons at the bottom of the **DCL Editor** window, choose Arrange Icons from the Window menu.

Getting Help

DCL provides extensive Help that describes how to use the graphical script building environment and documents each **DCL** statement, function, system variable, etc.

To display the Table of Contents for the **DCL** Help, choose Contents from the Help menu.

To search an index of **DCL** Help keywords, choose Search for Help On from the Help menu.

For information on Microsoft Windows Help, choose How to Use Help from the Help menu.

Context Sensitive Help

When you request context-sensitive Help, **DCL** displays a particular Help topic, based on your current activity. Context-sensitive Help is provided for each **DCL** language element, menu item, Toolbar button, and dialog box.




For help on a particular DCL language element

Select the language element (statement, function, or system variable) and press F1.



For help on a menu item (command)


1. Press Shift + F1 or choose the  button from the Toolbar.

The cursor changes to indicate that you are requesting context-sensitive Help.

2. Choose the menu and command you want Help on.



For help on a toolbar button

1. Press Shift + F1 or choose the  button from the Toolbar.
The cursor changes to indicate that you are requesting context-sensitive Help.
2. Choose the Toolbar button you want Help on.
For Help on the current dialog box, choose the Help button.

Text Editing Keys

You can use the following keys when exiting scripts:

Key	Description
Backspace	Delete the selection or delete the character preceding the cursor
Tab	Insert a tab character
Enter	Insert a new line, breaking the current line
CTRL+Insert, CTRL+C	Copy
SHIFT+Insert, CTRL+V	Paste
SHIFT+Delete, CTRL+X	Cut
Insert	Toggle between insert and overwrite typing modes
Delete	Delete the selection or delete the character following the cursor
Up	Move the cursor up one line
Down	Move the cursor down one line
Left	Move the cursor left one character position

Continued...

Key	Description
Right	Move the cursor right one character position
PgUp	Move the cursor up by one windowful
PgDn	Move the cursor down by one windowful
CTRL+PgUp	Scroll the window left by one windowful
CTRL+PgDn	Scroll the window right by one windowful
CTRL+Left	Move the cursor left by one word
CTRL+Right	Move the cursor right by one word
Home	Move the cursor to the start of the line
End	Move the cursor after the last character on the line
CTRL+Home	Move the cursor to the first character in the macro
CTRL+End	Move the cursor after the last character in the macro
SHIFT+Cursor Move	Drag the selection as the cursor moves
F1	Context-sensitive help for current command in editing window
SHIFT+F1	Context-sensitive help for current mouse cursor position.

