

Differences between Terrapin Logo 4 and Terrapin Logo

As much as we want Logo to work just like any other Logo flavor, we cannot reach this goal completely. The environments are too different, simply put. This chapter explains where Logo differs from Terrapin Logo. On the other hand, a browser environment provides rich support for Unicode character sets, various graphics formats, transparency, and more.

General

Since Logo runs in your browser, some commands are not available or are entirely different because the browser environment does not support them. The most notable commands are many commands related to file I/O; these work with the Files panel, and you must upload or download the resulting Zip archive manually.

Logo supports the extended Unicode character set including non-BMP characters with a value of up to 0xFFFFF. This includes all available Chinese, Korean or Japanese characters.

Drawing

You can now work with transparent colors! See [The section about colors](#) for further information.

Drawing is different from other Logo flavors because the background and the canvas are two separate panes. You draw on the canvas, which is transparent so the background can shine through. If you use a command like [SETBG](#) to alter the background, only the background is changed, and your drawing remains untouched.

Reading pixel data from the canvas is very difficult, especially because the drawing algorithms use anti-aliasing techniques to draw smoother curves. Therefore, the [DOT?](#) command reports TRUE for any pixel that is not fully transparent. Usually, you would expect the [DOTCOLOR](#) command to return the background color in areas where there is no drawing; here, [DOTCOLOR](#) returns the color [0 0 0], which is a fully transparent black color. You cannot use these two commands to examine the background, because it is not part of the canvas.

The turtle property TRANSPARENT is gone; BMP and JPG files can no longer be made transparent. If you need transparent images, use a format like GIF or PNG that supports transparency. Since Logo runs as a Web page, it can work with any kind of image that the browser can display.

[PENREVERSE](#) works differently from Logo 4.

Fonts

The list of available fonts that the [FONTS](#) command outputs is quite small, and fixed. Please let us know if you think that other fonts should be included. Also, the underline bit (4) is not longer supported, and will be ignored.

Input/Output

Access to disk files from within a Logo program is impossible; the browser keeps Web pages from accessing the local file system. Therefore, Logo has its own file system that you can upload or download as a packed Zip archive. A new panel, the Files panel, displays the contents of such a Zip archive file. You can also drag files there from outside the browser.

Apart from this file system, we have implemented a file system that links directly to DropBox, so you can use your DropBox account to load and save files, just like you used your local disk drive before. We are working on more of these file systems.

Two new commands, [READFILE](#) and [WRITEFILE](#) load and store Logo data in one chunk.

We have implemented full Unicode support, including the extended character sets for Chinese and other glyphs. Therefore, binary I/O using single byte values is not possible anymore.

Arrays

Arrays behave exactly like they used to, but they are not separate objects with their own properties anymore. Instead, arrays have been integrated into the Logo core. The [BYTEARRAY](#) has been removed altogether; use [ARRAY](#) instead.

[TEXTARRAY](#) has been expanded to output words, lists, and Unicode character codes.

You can now create arrays on the fly using curly brackets. This command creates an array of five elements:

```
MAKE "A {ONE TWO THREE FOUR FIVE}
```

Curly brackets can be nested to create multi-dimensional arrays.

Controls

All controls except for the [SCROLLBAR](#) control are present. That control has been replaced with the new [SLIDER](#) control. A new [BACKGROUND](#) property lets you set a background color, which is initially white. The [SIZE](#) property controls the size of the control as before, but negative values cause the control's auto-sizing feature to work again.

Controls can be rotated and scaled just like a turtle. Usually, this is not desirable, so a control's [LOCKSHAPE](#) property is initially set to [TRUE](#).

Button, check boxes, and radio buttons ignore a change of their [SIZE](#) property, because their size changes if the font or the label changes. Edit fields and popups ignore a change of their height; the size list boxes and static text fields can be changed freely.

List boxes and popups are no longer sorted by default; if you need a sorted list box, you will need to sort their elements by yourself. Thus, their callable [ADDSORTED](#) property is gone. The callable properties [APPEND](#) and [INSERT](#) also accept a list as input, causing each list element to become a separate list box entry.

A new control, the [GRID](#) control, lets you arrange bitmaps or other controls in a grid. Other new controls embed Web pages, videos, or a webcam.

Details about all properties can be found [here](#).

New Commands

Data Handling

[SPLIT](#) splits a word into a list, and [JOIN](#) converts a list into a word.

Drawing

[ALPHA](#) and [SETALPHA](#) report and set the alpha value for turtle drawings.

[BOUNDS](#) and [SETBOUNDS](#) report and set the viewport bounds, where turtles bounce off the bounds.

[BOUNCE](#) lets turtles bounce off the borders instead of stopping ([FENCE](#)) or wrapping around ([WRAP](#)).

[SHADOW](#) and [SETSHADOW](#) control a widget's drop shadow via its [SHADOW](#) property.

[OPACITY](#) and [SETOPACITY](#) control a widget's opacity.

Input/Output

[SETTEXTCOLOR](#), [SETTEXTFONT](#) and [TEXTFONT](#) let you set the font that you will be using for the next output commands, and the foreground and background colors for that font.

[READFILE](#) and [WRITEFILE](#) let you read and write Logo data in one big chunk.

The commands [GETCHAR](#), [PEEKCHAR](#) and [PUTCHAR](#) replace the old [GETBYTE](#), [PEEKBYTE](#) and [PUTBYTE](#) commands. They work with Unicode characters.

[FORM](#) formats a number.

Workspace

MILLISECONDS outputs the number of milliseconds since January 1, 1970. This command replaces the EXECTIME command.

Global Variables

:EPSILON contains the minimum value that two numbers may differ to still be considered to be equal.

:PROMPT gets and sets a prompt for the Input panel.

Properties

New **"PREFS** properties:

APPNAME	Contains the name of the Logo app if a Logo app is executing.
ARRAYBASE	This is the lowest value that you can use as an array or grid index. You may want to set the value to 1 if you prefer to index arrays and grid with a starting value of 1; actually, any starting value is legal as long as you are comfortable with that value.
CLASSROOM	Contains the name of the Logo classroom if Logo is running in a classroom.
STORAGE	Defines the storage used to load or save pictures or the Files Panel.

New widget properties:

BORDER	The border color; to make a border vanish, set it to a fully transparent color.
OPACITY	The opacity value.
SHADOW	The distance of a drop shadow.

New control properties:

COLOR	The text color.
-------	-----------------

Turtles have a new property:

ALPHA	The transparency value for drawings, a value between 0 and 1; 0 is fully transparent, and 1 is fully opaque.
-------	--

The **"LOGO.ENV** property list has new properties ENVIRONMENT, LAYOUT and SPLITTER.POS that gives you control over the look and feel of Logo.

Changed Commands

Global Variables

The setting of the global variable **:CASE** only controls whether the Logo parser converts quoted words to upper case. Text enclosed in vertical bars as well as back-quoted text is not affected.

:CURRENT.GRAPHICS is read-only because there is only a single Graphics panel.

Arrays

TEXTARRAY has been expanded to output words, lists, and Unicode character codes.

Turtles

[LOADSHAPE](#) (and [PLAY](#), for that sake) used to auto-lookup a shape or sound in the Toolbox. This has also been removed because of the time it takes to look up a file in the Cloud.

Input/Output

Previous Logo versions used a combined input/output field for printing and user input. This is a remnant from the era where user I/O mostly ran over teletype devices that printed on a sheet of paper. In a modern graphics environment, we do not have this paper-like input anymore. Instead, Logo prints to a text area, and the user enters commands and input in a separate edit field.

In the old times, the computer typed a prompt and then waited for user input. It does not make much sense to print the prompt into the output field and then have the user type in a response in the text field. Logo now has a new global variable `PROMPT` that lets you set a prompt.

[TYPE](#) does not insert any spaces when used in parentheses.

[KEY](#) reports characters and special words, not numbers anymore.

The old-style typelists that caused a Select File dialog to appear when used as a file name are still supported; we recommend the new list-style extension-lists. See [SELECT . FILE](#) for more information.

Menus

Menu handling has changed somewhat. Please see the [Menu Commands page](#) for details.

Properties

The `"PREFS` object's `USERNAME` property has changed its meaning.

The meaning of the following turtle property has changed:

Z. ORDER	The meaning has been reversed. A higher number means that a turtle is more in front. The lowest possible value is 1.
----------	--

Removed Commands

Arrays

`BYTEARRAY`, `BYTEARRAY?`

Input/Output

`CLEARINPUT`, `FILE.INFO`, `OPEN.PORT`, `GETBYTE`, `PEEKBYTE`, `PRINTQUOTE (PQ)`, `PUTBYTE`, `PUTBYTES`, `UNGETBYTE`, `RENAME`

Drawing

`VELOCITY/SETVELOCITY`, `TWINDOW/SETTWINDOW`

Workspace

`EXECTIME`

The aliases `ALL` and `C0` for the [CONTENTS](#) command have been removed. `C0` is now an alias for [CONTINUE](#), as it was before.

Global Variables

:BASE, :CURRENT.WINDOW, :DATADIR, :DESKTOPDIR, :LOGODIR

The properties of the "LOGO.ENV and "PREFS objects have changed as well. Many properties have been removed.

Properties

These turtle properties have been removed:

DRAGGABLE	You cannot drag widgets around anymore.
EXTENT	Turtle always move in pixel units.
SHOWHEADING	This debugging feature is gone.
STEPWISE	The scaling of turtle movements is currently not supported.
TRANSPARENT	Transparent BMP and JPG images are no longer supported. Please use GIF or PNG files with transparency information.