

MATLAB 5 Late-Breaking News

2 Getting Started

2 Installation

2 Supported Platforms

5 Help Desk

6 MATLAB Documentation

7 Platform-Specific Issues

7 PC

8 Macintosh

8 UNIX

10 API Characteristics That May Change in Future Releases

11 Known Software Problems

11 Language

12 Handle Graphics

13 Application Program Interface

14 PC-Specific Problems

16 UNIX-Specific Problems

17 Demos

18 Notes for MATLAB 5 Beta Users

18 Language Feature Changes

19 Graphics Feature Changes

21 Application Program Interface (API) Feature Changes

22 UNIX Path Information

Getting Started

Installation

Install your copy of MATLAB® 5 into an empty directory. Do not install over a previously installed version of MATLAB.

See the “UNIX” section on page 8 for information about installing MATLAB on UNIX platforms.

Supported Platforms

PC

MATLAB for the PC (Microsoft Windows) is designed to run in 386-enhanced mode on any IBM, Compaq, or 100% compatible system with an Intel 486, Pentium, or Pentium Pro processor running Microsoft Windows NT 3.51 or 4.0 or Windows 95. MATLAB for the PC is a full 32-bit application.

In addition to Microsoft Windows NT or Windows 95, this version of MATLAB requires the following minimum configuration:

- 387 or 487 math coprocessor chip (except 486DX processors, which include this internally)
- Microsoft Windows-supported CD-ROM drive (for installation)
- High-density 3 1/2" floppy disk drive (for installation)
- Microsoft Windows-supported mouse
- Microsoft Windows-supported monitor
- 8-bit graphics adaptor and display (for 256 simultaneous colors)
- 8 MB RAM (for Windows 95) or 12 MB RAM (for Windows NT)

See the *Installation Guide* for details about the hard disk space requirements.

The following items are strongly recommended

- Additional memory (to bring the total to at least 16 MB)
- Microsoft Windows-supported graphics accelerator card
- Microsoft Windows-supported printer

- Microsoft Windows-supported sound card
- Microsoft Word 6.0 or 7.0 (to use the MATLAB Notebook)
- One of these compilers for building MEX-files: Microsoft Fortran PowerStation 4.0 or Visual C/C++ 4.0 (no additional linker needed), Borland C/C++ 4.5 (no additional linker needed), or Watcom C/C++ 10.6 (no additional linker needed)

UNIX

Minimum system resources:

- 40 MB disk space
- 16 MB memory
- 64 MB swap space

Other system requirements depend on the particular UNIX platform on which you are running MATLAB.

Sun SPARC (SunOS 4)

- SPARC-based workstation
- SunOS 4.1.4
- Open Windows version 3.0 or X Windows (X11R5)

Sun SPARC (Solaris 2)

- SPARC-based workstation
- Solaris 2.5 or later (SunOS 5.x)
- X Windows (X11R5)

HP 9000

- HP 9000 PA-RISC workstation
- HP-UX 10.01 or later
- X Windows (X11R5)

DEC Alpha

- DEC Alpha workstation
- Digital UNIX 4.0 or later
- DECwindows or X Windows (X11R5)

IBM RS/6000

- IBM RS/6000 workstation
- AIX 4.1
- X Windows (X11R5)

Silicon Graphics (SGI)

- SGI (R4000) MIPS-based workstation
- IRIX 6.2 or later
- X Windows (X11R5)

Silicon Graphics (SGI64) (Beta Software)

- SGI (R8000/R10000) MIPS-based workstation
- IRIX 6.2 or later
- X Windows (X11R5)

Linux

- 80486 or Pentium PC
- Linux 2.0.18 kernel (Red Hat 4.0 distribution)
- X Windows (X11R6)

Macintosh

MATLAB for the Macintosh is distributed on CD-ROM only. MATLAB for the Macintosh is designed to run on:

- Any Power Macintosh
- Any Macintosh equipped with a 68020 or 68030 microprocessor and a 68881 or 68882 math coprocessor
- Any Macintosh equipped with a 68040 microprocessor (the math coprocessor is built in)

MATLAB will not run on a Macintosh with a 68LC040 microprocessor.

The Macintosh must have an 8-bit graphics display and, for installation, a CD-ROM drive and a high density 1.4 MB 3 1/2" floppy disk drive.

To run the MATLAB Notebook, you need Microsoft Word 6.0 and 8 MB additional RAM (for Power Macintosh) or 3 MB RAM (for 68000 series).

See the *Installation Guide* for details about the hard disk space requirements and the supported compilers.

In addition, the following system configuration is recommended for this version of MATLAB:

- 26 MB of free space on your hard drive (34 MB for MATLAB with SIMULINK). An additional 60 MB is required for the optional online help system.
- 16 MB memory partition for MATLAB
- Apple LaserWriter or other PostScript printer
- System 7.1 or later (System 7.5 or later is preferred)

Help Desk

The MathWorks Help Desk is an enhanced Help facility that provides access to online help topics, online reference materials, electronic documentation, and World Wide Web pages through a Web browser. You do not need to be connected to the Internet to use this facility.

Windows and Macintosh users can access this facility via the **Help** menu or the ? icon on the Command Window toolbar. Users on all platforms can access this facility via the `hel pdesk` command.

Platform Issues

Currently the Help Desk should be used with Netscape Navigator Release 2.0 or Microsoft Internet Explorer 3.0 (the Help Desk does not work as well with earlier releases of those products).

When invoked from Microsoft Internet Explorer, the index search facility generates HTML output that takes the place of the Help Desk page in the browser's page hierarchy. Consequently, the **Back** button or hyperlinks to the Help Desk will not work as expected. To return to the Help Desk from the index search page, use the **Refresh** button.

Also, the search may not work correctly when invoked the first time over a slow network; subsequent searches will work correctly.

Note the Java JIT compiler must be disabled to run the index search applet under Internet Explorer. To turn off JIT, click the **View** menu and then click **Options**. Select the **Advanced** tab, and then deselect the check box labeled **Enable Java JIT Compiler**.

On the 68K Macintosh, the index search facility requires Netscape 3.0. In addition, the index search facility is extremely slow on the 68K Macintosh.

MATLAB Documentation

The MATLAB documentation set has been rewritten, expanded, and divided into several volumes for ease of use. The set currently consists of online help, as well as hypertext-based manuals.

If you are a new MATLAB user, we recommend you start by reading *Getting Started with MATLAB*, which explains how to get started with the fundamentals of MATLAB.

If you are upgrading from MATLAB 4, we recommend you start by reading *MATLAB New Features Guide*, which introduces the new features of MATLAB 5 and describes how to upgrade MATLAB 4 applications for use with MATLAB 5.

Online help and demos provide online reference information about MATLAB commands and demonstrate some of MATLAB's features.

Platform-Specific Issues

PC

See “PC-Specific Problems” on page 14 for additional information about known MATLAB software problems on PC platforms.

Printing Under Microsoft Windows

Before you can print from a Microsoft or Novell NetWare network environment under Windows 95 or NT, you must map the LPT1 port to the printer you want to use.

To map LPT1 on Microsoft networks, issue this command at the system’s command prompt:

```
net use LPT1: \\server\printer
```

where *server* is the name of the server sharing the printer and *printer* is the name of the printer.

On Novell NetWare networks, use this command:

```
capture l=1 q=printer
```

where *printer* is the name of the print queue.

If you are using a Microsoft network, you can map LPT1, or you can edit the `prn topt` function to change the definition of `pcmd` to:

```
COPY /B %s \\server\printer:
```

where *server* is the name of the server sharing the printer and *printer* is the name of the printer.

Problems Printing Using Z-Buffer If your system takes an excessively long time to print Z-buffer figures, you may need to switch to painters model.

PC-Specific Characteristics of MATLAB That May Change

This section lists some specific characteristics of MATLAB for the PC that are likely to change in future releases.

- `drawnow 'discard'` is not working.
- The `print` command redraws the figure each time it is called.
- Path information now resides in `pathdef.m` instead of in `matlabrc.m`.
- You cannot place a frame object on top of a `Uicontrol` object.

Also note that MATLAB 5 running with Microsoft Windows NT requires NT version 3.51 (not 3.5).

Macintosh

Additional Information

See the `README` file for additional information concerning MATLAB for Macintosh.

UNIX

Using `license.dat` for UNIX Installations

To expedite the installation of the software, we have included the `license.dat` file needed to install the software on the CD-ROM.

Copy this file from the CD-ROM into the file `$MATLAB/etc/license.dat` before proceeding with your installation.

UNIX Installation Messages

- When you install MATLAB on any HP700 or SGI workstation, the installation succeeds, but you may notice a message in the shell window at the end of the installation containing the following line:

```
Tar: blocksize = 16
```

You may safely ignore this message.

- When you install MATLAB on an HP700 running the HP-UX 10.01 operating system, the installation succeeds, but several error messages appear in the shell window at the end of the process:

```
A fatal error occurred while running 'xsetup' the X Window System version of 'install'. The following error was returned by this program:
```

```
Tar: blocksize = 16
```

```
X Error of failed request: BadMatch (invalid parameter attributes)
```

```
Major opcode of failed request: 42 (X_SetInputFocus)
```

```
Serial number of failed request: 12029
```

```
Current serial number in output stream: 12030
```

You may safely ignore these messages.

API Characteristics That May Change in Future Releases

This section lists some specific characteristics of the MATLAB 5 API (Application Program Interface) that are likely to change in future releases.

- In MATLAB 5, MATLAB arrays can share data. There is currently no way for a MEX-file to determine that an array contains shared data. MEX-files that modify their input arguments may corrupt arrays in the MATLAB workspace. This style of programming is strongly discouraged.
- The Fortran interface to the new MATLAB API routines has not yet been implemented.
- Engine support is now implemented on PC platforms using OLE V4 data types only.
- In MATLAB 5, changing directories unloads any loaded MEX-files and runs any registered `mexAtExit` routines. This was not true of MATLAB 4. You can override this behavior by calling the new MEX API routine, `mexLock()`. This allows you to lock the MEX-file in memory. MEX-files that are locked should be on the MATLAB path. If, after changing directories, a locked MEX-file is no longer on the MATLAB path, it will not be unloaded but can no longer be executed.
- `mexIsLocked` currently *always* returns the value 1.

Known Software Problems

This section describes MATLAB 5 known software problems, providing workarounds for most problems.

Language

Unintended Array Reshaping

MATLAB 5 allows you to pass arrays with more than two dimensions to certain functions that are not designed to take an argument of this kind.

For example, you can supply a three-dimensional array as an argument to `plot`, but it treats the data as a two-dimensional array for plotting. Passing `plot` a 3-by-3-by-2 array produces the same result as passing `plot` the same array reshaped to 3-by-6. In this case, the shape of the original array in the workspace (3-by-3-by-2) is unchanged. However, if the first dimension of the array that you pass to `plot` is 1, `plot` *reshapes the original array in the workspace* to a row vector with the same number of elements. If `A` is a 1-by-3-by-2 array, typing `plot(A)` causes `A` to be reshaped to a 1-by-6 vector in the workspace, which may lead to errors in subsequent calculations. If this occurs, you can return `A` to its original dimensions with the `reshape` command.

In general, avoid passing multidimensional arrays to functions that are not expecting them.

Using `==` with an Empty Matrix

The expression `A == []` produces a scalar (1-by-1) value in MATLAB 5, although the other Boolean operators (`>`, `<`, etc.) yield an empty matrix, `[]`. MATLAB produces the following warning message when this expression is used:

```
Warning: X == [] is technically incorrect. Use isempty(X) instead.
```

Debugging with dbstop

dbstop returns a Can't find M-file function error if you include the .m extension on the name of the M-file when you type the command. To avoid this error, leave off the .m extension when you type the command. For example, type dbstop myfile, not dbstop myfile.m.

Handle Graphics

Note that there are some additional Handle Graphics® issues discussed in the section “PC-Specific Problems” on page 14 and “UNIX-Specific Problems” on page 16.

Printing GUIs

GUIs that have a very large number of Uicontrols may take a long time to print when you use the print -dwin command.

Additionally, if you open another application on top of the GUI window while printing is in progress, elements of the top window may appear in the printout. To avoid this, wait until MATLAB indicates printing is finished before opening another window on top of your GUI.

Graphics on Black and White Monitors

On all platforms except Macintosh, if you use a black and white monitor, you will see the following message when MATLAB opens:

```
Warning: uimenu BackgroundColor is non-functional, and will  
be removed in a future release.
```

By default, graphics functions plot white lines on a white background, which are not visible on a black and white monitor.

To eliminate the warning and correct the plotting problem, edit the M-file matlabrc.m using the M-File Editor/Debugger (or another editor) so that lines 62-70 look like the code shown on the next page. Note that lines 65, 67, and 68 in the original are changed, and line 74 in the original is moved to line 63 in the edited code.

```

% Make ui controls, uimenu and lines look better on
% monochrome displays.
colordef(0, 'white') % Set up for white defaults
if get(0, 'ScreenDepth')==1,
    set(0, 'DefaultUIControlBackgroundColor', 'white');
% set(0, 'DefaultUIMenuBackgroundColor', 'white');
set(0, 'DefaultAxesLineStyleOrder', '-|--|:|-.');
set(0, 'DefaultAxesColorOrder', [0 0 0]);
set(0, 'DefaultFigureColor', [1 1 1]);
end

```

A second way to correct the plotting problem is to add the following lines to your startup. m-file. This will not remove the warning, however.

```

set(0, 'DefaultAxesLineStyleOrder', '-|--|:|-.');
set(0, 'DefaultAxesColorOrder', [0 0 0]);
set(0, 'DefaultFigureColor', [1 1 1]);

```

dragrect and rbbox Functions

The `dragrect` function assumes that rectangles passed to it are specified in pixel units, not current Figure units. To avoid incorrect scaling of the rectangle(s) when using `dragrect`, make sure that arguments are specified in pixel units.

In function `rbbox(initialRect, fixedPoint, stepSize)`, the `stepSize` argument is in pixel units. Arguments `initialRect` and `fixedPoint` are in the current Figure units. Under Linux, the `cxopts.sh` file does not work with the latest version of `gcc` (2.7.2). To correct the problem, change the value of `LD` to `gcc`, and edit the value of `LD_FLAGS` to remove the flag `-rdynamic`.

Application Program Interface

MATLAB Compiler

MEX-files generated from M-files using the MATLAB 4 Compiler do not run in MATLAB 5, and cannot be recompiled with the `-V4` switch to the `mex.m` script. You will need to regenerate your MEX-files using the MATLAB 5 Compiler. This product was not yet available at the time of printing. Please contact The MathWorks, Inc. for availability.

Assertions

The documentation for the `mxAssert` macro indicates that assertions are *off* by default, and are activated when MEX-files are compiled with the `-g` flag. Currently, however, assertions are *on* by default.

To remove assertions, add the `-DNDEBUG` flag in the call to `mex` (this is then passed on to the compiler). To turn assertions off by default, append the `-DNDEBUG` flag to the `COPTIMFLAGS` string in the `mexopts.sh` shell script.

PC-Specific Problems

The following problems apply to Microsoft Windows 95 and Windows NT platforms running MATLAB, unless otherwise indicated.

Launching the M-File Editor/Debugger without MATLAB

When you launch the M-File Editor/Debugger without MATLAB open, the following warnings are displayed: Unable to connect to MATLAB and Error in evaluating an expression. The editor will open as usual when you dismiss these dialog boxes.

To avoid the warnings, have MATLAB open when you launch the M-File Editor/Debugger from Microsoft Windows, or start the M-File Editor/Debugger from within MATLAB.

Folder and Filenames with Spaces

MATLAB produces an error when you try to open an M-file using **Open...** in the Command Window **File** menu (or using the toolbar shortcut), if there are spaces in the filename or in the name of any folder in the file's path. To open files with spaces in their names or in their paths, either open them from within the M-File Editor/Debugger, or open them from Microsoft Windows (see the previous problem description).

Managing Paths

In the Path Browser, the **Restore Defaults...** button may not function properly if the `pathdef.m` file becomes sufficiently corrupted.

In this case, you can restore the defaults from the file `pathdeforig.m` in the `matlab\toolbox\local` directory. First, identify whether the corrupted `pathdef.m` is in the `matlab\toolbox\local` directory or in the

MATLAB startup directory, by typing `whi ch pathdef` at the command line when you experience the problem. Then simply rename the offending `pathdef. m` (as `pathdef. ol d`, for example) and *make a copy* of `pathdefori g. m` from `matlab\toolbox\local` in the same location. Rename this copy `pathdef. m` and restart MATLAB.

Browsing Paths

When you click on the **Browse...** button in the Path Browser, the **Change Current Directory** dialog may not default to the current directory. If you have a path highlighted in the **Path** listing when you click **Browse...**, that path is displayed in the **Change Current Directory** dialog. To start browsing from your current directory, click in the **Current Directory** box before clicking the **Browse...** button.

Uicontrol Background Color

You cannot change the background color of a Uicontrol.

GUIDE on Windows NT 3.51

On computers running Windows NT 3.51, GUIDE (Graphical User Interface Development Environment) may leave extra menus (**Options**, **Tools**) visible on the figure window when you close it.

Line Width on Windows 95

In MATLAB for Windows 95, the line width of a nonsolid line cannot be changed.

MS-Windows TCP/IP Issues

MATLAB 5 for Microsoft Windows 95 requires the TCP/IP networking software that came on your Windows 95 CD to be installed on your computer. Type `hel p mi pc` at the command line for information about installing TCP/IP from your Windows CD.

If you start MATLAB without TCP/IP installed, you will see the warning `Unabl e to i ni ti al i ze MI PC. ,` and certain components of MATLAB will not function properly.

If your computer does not have networking software installed at all (for example, your computer is not networked and does not have a modem) you may also need to install Dial-Up Networking from the Windows 95

CD. If you continue to experience MIPPC-related warnings after installing TCP/IP on your computer, install Dial-Up Networking:

- 1 Insert the Windows 95 CD-ROM.
- 2 Click **Add/Remove Software** when the **Windows 95 CD-ROM** window opens.
- 3 In the **Add/Remove Programs Properties** window, click the **Windows Setup** tab.
- 4 Click on **Communications** in the list to highlight it. Do not click the check box.
- 5 Click on **Details...**
- 6 Click the check box next to **Dial-Up Networking**.
- 7 Click **OK** in the **Add/Remove Programs Properties** window.
- 8 Remove the Windows 95 CD-ROM.

UNIX-Specific Problems

Printing with Ghostscript Drivers

The `print` command does not currently work using Ghostscript drivers on the HP700, DEC Alpha, and IBM RS6000 platforms.

Uimenu on X-Windows

On X-Windows, a Uimenu may not function properly if you set the `'enable'` property to `'on'` when it was already `'on'`, and a modal figure is visible at the time. (A modal figure is a figure that has its `'WindowStyle'` property set to `'modal'`, such as a dialog box that requires a user response before allowing access to other MATLAB windows.)

cssopts.sh File on Linux

Under Linux, the `cssopts.sh` file does not work with the latest version of `gcc` (2.7.2). To correct the problem, change the value of `LD` to `gcc`, and edit the value of `LDFLAGS` to remove the flag `-rdynamic`.

Demos

Vibration Movie Demo

In the Vibration movie demo, MATLAB produces an error message if you press the **Info** button *after* making the movie (by pressing the **Make** button). To view the information, press the **Info** button *before* making the movie.

Command Line Demos

Some command line demos (like `imagedemo`) leave a Figure window open on the screen when they finish running. You can close the window as you normally would.

The Makevase Demo

The `makevase` command line demo can cause a MATLAB segmentation violation when the **Close** button is pressed while the shape is being drawn. This error occurs if the **Close** button is pressed *after* the **New Shape** button is pressed, and *before* the last point of the shape is placed. To avoid this error, complete the shape (by clicking with the right mouse button or **Shift**-clicking) before pressing the **Close** button.

Notes for MATLAB 5 Beta Users

Read this section only if you have worked with Alpha or Beta versions of MATLAB 5, and are now working with the FCS (First Customer Ship) version of MATLAB 5.

This section describes features and characteristics of MATLAB 5 that were introduced during the MATLAB 5 Alpha/Beta program that have changed either during the MATLAB 5 Alpha/Beta program or in the MATLAB 5 FCS version.

Language Feature Changes

argname Function Changed to inputname

The `argname` function has been renamed to the `inputname` function.

bittest Function Renamed bitget

The `bittest` function has been renamed `bitget`.

bug Function Removed

The function `bug`, which during the MATLAB Alpha/Beta program enabled you to create a bug report, has been removed for FCS. Use the Help Desk, as described on page 5, to report bugs.

extract Function Removed

The function `extract`, which extracted cell array contents, has been removed. Use direct cell array indexing with the `{ }` notation to access the contents of an array.

fields.m File Renamed fieldnames.m

The M-file `fields.m` has been renamed `fieldnames.m`.

numeric Function Removed

The `numeric` function has been removed. Use `double` to convert a character into its numeric codes.

rightjust Function Renamed to strjust

The function `rightjust`, which right-justifies a character array, has been renamed `strjust`.

structs Function Combined into struct

The function `structs`, which creates structures, has been combined into `struct`.

Graphics Feature Changes

Axes Stretch Property Removed

The `Axes Stretch` property has been removed. MATLAB now disables the stretch-to-fill behavior whenever the `Axes DataAspectRatioMode`, `PlotBoxAspectRatioMode`, or `CameraViewAngleMode` is set to `manual`. Note that specifying a value for the `DataAspectRatio`, `PlotBoxAspectRatio`, or `CameraViewAngle` sets the associated mode to `manual`.

Axes ViewPortScale and ViewPortScaleMode Properties Removed

The functionality that was embodied in the `ViewPortScale` property is now achievable using the `CameraViewAngle` property. Note that setting `ViewPortScale` to 1 is equivalent to using the new `zoom fill` option.

ChangeFcn and CurrentProperty Properties Removed

The `ChangeFcn` and `CurrentProperty` properties have been removed.

bar and barh Functions Changed

The `bar` and `barh` functions no longer support the two output argument forms that they did during the MATLAB 5 Alpha/Beta program:

```
[xb, yb] = bar  
[xb, yb] = barh
```

colordef Function Replaces Several Functions

The `colordef` function replaces the `kdefault`, `wdefault`, `default4`, and `cleardef` functions.

ExecutionQueue Property Removed

The `ExecutionQueue` property has been removed. The `BusyAction` property provides similar functionality.

See the `axes` function in the online MATLAB Graphics Reference for more information about each property.

HandleVisibility Property Replaces HiddenHandle Property

The `HandleVisibility` property replaces the `HiddenHandle` property.

`HandleVisibility` is a property of all objects. It controls the scope of handle visibility within three different ranges. Property values can be:

- `on` — The object's handle is available to any function executed on the MATLAB command line or from an M-file. This is the default setting.
- `callback` — The object's handle is hidden from all functions executing on the command line, even if it is on the top of the screen stacking order. However, during callback routine execution (MATLAB statements or functions that execute in response to user action), the handle is visible to all functions, such as `gca`, `gcf`, `gco`, `findobj`, and `newplot`. This setting enables callback routines to take advantage of the MATLAB handle access functions, while ensuring that users typing at the command line do not inadvertently disturb a protected object.
- `off` — The object's handle is hidden from all functions executing on the command line and in callback routines. This setting is useful when you want to protect objects from possibly damaging user commands.

For example, if a GUI accepts user input in the form of text strings, which are then evaluated (using the `eval` function) from within the callback routine, a string such as `'close all'` could destroy the GUI. To protect against this situation, you can temporarily set `HandleVisibility` to `off`:

```
user_input = get(editbox_handle, 'String');  
set(gui_handles, 'HandleVisibility', 'off')  
eval(user_input)  
set(gui_handles, 'HandleVisibility', 'commandline')
```

-psdefcset Option for print Function Renamed -adobecset

The `-psdefcset` option to the `print` command has been renamed to `-adobecset`. This option specifies that MATLAB should use Adobe's default character set encoding for PostScript, rather than ISO Latin-1 character set encoding.

Zbuffer Property Removed

The `ZBuffer` Figure property has been removed. Statements that instruct you to set the `ZBuffer` property should be replaced with statements that set the `Renderer` Figure property. For example, replace:

```
set(gcf, 'ZBuffer', 'on')
```

with

```
set(gcf, 'Renderer', 'zbuffer')
```

Application Program Interface (API) Feature Changes**mexSearchMATLABPath Function Removed**

The function `mexSearchMATLABPath` has been removed from the MEX interface. Replace `mexSearchMATLABPath()` with a call to the `MATLAB which` function using `mexCallMATLAB()`.

mxCreateDoubleMatrixFromData Function Removed

The function `mxCreateDoubleMatrixFromData` has been removed from the MEX interface. Replace `mxCreateDoubleMatrixFromData` with a call to `mxCreateDoubleMatrix`, followed by copying the data into the array.

mxReshape and mxGetSize MEX-functions Renamed

The following MEX functions that were introduced in the MATLAB 5 Alpha/Beta program have been renamed:

- `mxReshape` is now `mxSetDimensions`.
- `mxGetSize` is now `mxGetDimensions`.
- `mxIsGlobal` is now called `mexIsGlobal`. Rewrite and recompile MEX-files that call `mxIsGlobal`.

UNIX Path Information

Path information now resides in `pathdef.m` instead of in `matlabrc.m`.