IDL 8.3 Release Notes

Revised 23 October 2013

* [New Routines and Features](http://ittvisual.securehttp.internapcdn.net/secure_ittvisual/envi_5.1/IDL_release_notes.html?token=445078872806#new)
* [Known Issues](http://ittvisual.securehttp.internapcdn.net/secure_ittvisual/envi_5.1/IDL_release_notes.html?token=445078872806#knownall)
* [Fixed Issues](http://ittvisual.securehttp.internapcdn.net/secure_ittvisual/envi_5.1/IDL_release_notes.html?token=445078872806#Problems_Fixed_in_IDL_8.3_)

Please visit the Help Articles section of the Exelis website (www.exelisvis.com) for the most up-to-date release notes.

New Routines

For additional information on new routines in the IDL 8.3 release, see the "What's New" topic in IDL Help.

* The IDLUNIT function allows you to perform basic dimensional analysis, conversions, and mathematical operations, all while factoring units.
* The LAMBERTW routine allows you to compute the value of the Lambert W function.
* IDL now has three new HDF5 wrapper routines to simplify the process of reading and writing data: H5\_GETDATA, H5\_LIST, and H5\_PUTDATA.
* The SIGNUM function returns the sign of each element of an array.
* The IDLUNIT function allows you to perform basic dimensional analysis, conversions, and mathematical operations, all while factoring units.

New Features

For additional information on new features in the IDL 8.3 release, see the "What's New" topic in IDL Help.

* IDL introduces support for the JPEG 2000 Interactive Protocol (JPIP) via the IDLnetJPIP Object Class.
* Implied Print allows you to enter an expression or a variable name at the IDL command prompt and get a result rather than throwing a syntax error.
* Asynchronous timers provide IDL with non-blocking timers.
* You can now create static methods on IDL object classes and use the standard "dot" syntax to make static method calls.
* The new colon ":" operator can be used to construct arrays with increasing (or decreasing) values.
* The PLOT, CONTOUR, and SURFACE functions have a new Equation argument (and EQUATION property) that lets you dynamically view your data.
* IDL now contains the DICTIONARY and ORDEREDHASH data types, based upon the HASH class.
* The IDLnetURL class now has a ::Delete method which allows you to delete a file on an HTTP server.
* IDL introduces support for the JPEG 2000 Interactive Protocol (JPIP) via the IDLnetJPIP Object Class.
* The CW\_FIELD widget has four new keywords: DOUBLE, FOCUS\_EVENTS, IGNORE\_ACCELERATORS, ULONG.
* The IDL command line will now ignore "IDL>" within pasted commands.
* You can now set !DLM\_PATH or the IDL\_DLM\_PATH preference and IDL will immediately register any .dlm files that it finds within the new directories. This allows you to easily add new C-code routines to IDL without having to restart your IDL session.
* The IDLgrContour object now has a new LABEL\_THRESHOLD property that lets you control whether short contour line segments are labeled or not.
* In the IMAGE function, you can now pass a string array of color names to the RGB\_TABLE property.
* The graphics TEXT function now has a new ONGLASS property that lets you display the text either within the plane of the screen, or using the full three-dimensional transform.
* The graphics routines now have the option to not draw the axes,  but to preserve the same margins as if the axes were there. Set AXIS\_STYLE=4 to use this new feature.
* Graphics windows now have the option to zoom in a single axis dimension.
* Event handling while debugging: IDL sends widget events even when stopped within a routine. Set the system variable  !DEBUG\_PROCESS\_EVENTS = 0 to disable this behavior.
* If you are using TIC and TOC to profile your code (with the PROFILER keyword), and you are using the IDL Workbench, then IDL will automatically refresh the Profiler View when TOC is called.

Backward-Compatibility Issues in IDL 8.3

Change request ID numbers are included where applicable.

All Platforms

|  |  |
| --- | --- |
| **ID** | **Description** |
| **n/a** | **Code SAVE Created with IDL 8.3 Cannot be Restored in Previous Versions of IDL.** The IDL Save file revision level was increased from 11 to 12 because of the new colon operator (see the "What's New in IDL 8.3" Help topic). Save files that contain IDL code and that are created with IDL 8.3 cannot be restored in previous versions of IDL. If you want to distribute a code SAVE file to other users who may only have IDL 8.0, IDL 8.1, or IDL 8.2, you should create your SAVE file using one of these earlier IDL versions.  **Note:** This change does not affect older Code SAVE files or SAVE files that contain data. These files can still be restored in IDL 8.3. |
| **n/a** | **IDL Workspace Migration** IDL does not automatically update the IDL workspace between releases. After upgrading to a new IDL version, you have the following options:   1. Create a new workspace to use with the new IDL version. You will need to reset any IDL Workbench preferences you may have set in the old version. 2. Continue using the old workspace without modification. New views may not be available. 3. Use the **Window > Show View > Other** menu option to update the workspace to specify the views you want to display. |
| **n/a** | **IDL Core Preference Migration** IDL does not automatically migrate IDL core preferences between releases. After upgrading IDL to a new version, you should:  Reset any IDL core preferences you want to use with the new IDL version by using the Preferences dialog or the PREF\_SET routine. |
| **n/a** | **Additional Documentation is Provided In Acrobat (PDF) Documents** The IDL 8.3 online help contains all the routines and functions in the IDL language, including the new Graphics routines and examples. Documentation that does not exist in the online help is provided in fully searchable PDF format as part of the IDL installation, located in <install\_dir>\help\pdf. |

Windows

None

Known Problems in IDL 8.3

Change request ID numbers are included where applicable.

All Platforms

|  |  |
| --- | --- |
| **ID** | **Description** |
| **IDL-42545** | **WIDGET\_TREE Drop Event Can Cause IDL to Crash if an Error Exists in the DRAG\_NOTIFY Callback Routine** If there is an error (such as an undefined variable) in the custom DRAG\_NOTIFY callback routine, IDL may freeze or crash since the error halts execution in the child routine and IDL cannot return to the calling function.  **Workaround**: Handle potential problems in the callback routine by including CATCH statements in your code. See the documentation topic "Responding to Drag Notifications (Callbacks)" for more information. |
| **IDL-55393** | **Breakpoints in Files Opened via Links, Shortcuts, or Alternate Paths** To improve performance, IDL always compiles the first version of a file it finds in its path. If you open a file in the editor and set a breakpoint, then compile from a version of the same file with a different path, IDL will not assume that the two files are the same. As a result, IDL will not stop at the breakpoint set in the second file. Alternate paths to the same file can be constructed via UNIX links, Windows shortcuts, mapped drives, or UNC paths. **Workaround:** Make sure that the full path to the file in which you set the breakpoint is the same as the full path to the compiled file. If you use the IDL Workbench, compiling using the Workbench menus or toolbars will ensure that the same path is used for compilation and debugging. |
| **IDL-49236** | **Deleting a Project or a Directory Within a Project While the Project Is Open Causes Exceptions when the Workbench Exits** When a project file or a directory within a project is deleted from outside the Workbench (for example, from Windows Project Explorer) while the project is open, a warning message may be issued while the Workbench is still running. An exception may occur on exiting the Workbench. |
| **IDL-51318 IDL-51320 IDL-51322** | **Formatter Preferences for Keywords, System Variables, and Structure and Class Names are Ignored by the Source→Format Command** The keyword, system variable, and structure and class name formatting preferences set on the **IDL > Formatter** preference page are not applied when the code is highlighted and the **Source > Format** command is selected. |
| **IDL-58532** | **Default UTF-8 Encoding is Corrupting .pro Files with Non-English Characters** Non-English characters are converted to symbols when imported into the IDL Workbench. Importing a .pro file containing non-English characters in IDL permanently corrupts the file with bad characters. **Workaround**: Before opening a file or before saving an open file, change the file encoding to ISO-8859-1 for Spanish or German, or to Shift-JIS for Japanese.  To change the encoding for an individual file, open the file in an IDL editor, then select **File > Properties**. In the Resources dialog box, change the 6**Text file encoding**preference.  To change the default encoding preference for IDL, select **Window > Preferences**. In the Preferences dialog box, choose **General > Workspace**. Change the **Text file encoding** preference. |
| **IDL-57214 IDL-58285 IDL-58839** | **IDL Crashes on Macintosh and UNIX When Performing Graphics Operations Where GL Drivers are Present** Possible workaround: Rename the gl\_driver.so file (located in bin\bin.<*platform>*) to gl\_driver.ba. |
| **IDL-60473** | **Accessing Preferences while IDL is Busy May Hang Workbench** Trying to change Preferences while IDL is busy (for example, when a lengthy command is being executed) may cause the Workbench to hang. |
| **IDL-61842** | **IDL on Windows 7 May Have Unpredictable Behavior When Using Graphics** The cause may be graphics cards that are unsupported in Windows 7. Please see a list of supported graphics cards on the Microsoft.com website. You may want to upgrade your graphic driver or switch the IDL Graphics preference to use software rendering. |
| **IDL-63902 IDL-52705** | **Using the IDL\_Java Bridge Causes the Workbench to Ignore the Current Directory in CLASSPATH  Workaround**: Copy the .class file to any other directory explicitly listed in CLASSPATH. |
|  |  |

Macintosh Platforms

| **ID** | **Description** |
| --- | --- |
| **IDL-60599** | **On Macintosh OS, Only One Workbench Instance Can Run at a Time** Even if the preference "Prompt for workspace on startup" is set and multiple workspaces are available, only one instance will run. **Workaround:** Run the Workbench from the init script located in the idlde.darwin.x86\_64.app/Contents/MacOS directory of your IDL installation. |
| **IDL-64149** | **Widget Interface Loses Focus When Selection is Made in a Drop-down List  Workaround:** In X11 Preferences, enable "**Click-through Inactive Windows**" |
| **IDL-67319 IDL-67320** | **Graphics Rendering Causes Memory and Device Errors** XQuartz version 2.7.2 causes these error messages when creating IDL Graphics. When IDL is installed, the Macintosh operating system may direct you to install the most recent version of XQuartz. **Workaround:** Use the latest version of XQuartz. |
| **IDL-68845** | **Launching the IDL Workbench on Macintosh May Prompt Installation of Java v1.6** If you launch the IDL Workbench on Macintosh without first having Java v1.6 installed, the Macintosh OS may prompt you to install it first. This will happen even though IDL includes Java v1.7. For more information on this issue, please see: <http://stackoverflow.com/questions/11919004/eclipse-4-2-mac-os-x-10-8-ml-and-java-6>  **Workaround:** Install the Macintosh version of Java 1.6 when prompted. The Workbench will continue to use Java v1.7 as it was installed with IDL. Another workaround is to add a softlink as sudo from Frameworks 1.6 -> 1.7 as described in the link above. |

Windows Platforms

|  |  |
| --- | --- |
| **ID** | **Description** |
| **IDL-27165** | **Using DIALOG\_PICKFILE with the MULTIPLE Keyword Limits the Maximum nFiles or Maximum nBytes for a File Array** When attempting to open multiple files using DIALOG\_PICKFILE, the operating system has a maximum buffer of 32767 bytes. |
| **IDL-47445** | **Export Bridge Instantiation Object Creation Failure with Embedded License** Attempting to create an export object with an embedded license fails when no printer is set up. **Workaround**: Install and set up a printer before running IDL. |
| **IDL-48454** | **WIDGET\_TIMER Can Cause Applications To Freeze Until An Event Is Sent To The Application** WIDGET\_TIMER events fail after 10 events. This problem happens in the following cases:   * Timer events are chained together by setting the next timer within the handling of the current timer and before the "work" section of the event handler is performed. * The next timer goes off during the current timer event processing (for example, during a lengthy calculation).   **Workaround:** Chain timer events by setting the next timer after any computations are completed. For smoother performance, set the next timer based on how long the computations execute, which accounts for variances in machine speed. For example, to fire the timer once every second, set the next timer to 1.0 - (*time to perform computation)* |
| **IDL-49641** | **Updating Edited Variable Values is Slow While IDL is Busy or Blocking** If you edit a variable value using the IDL Workbench Variables view while IDL is busy (for example, when a lengthy command is being executed, or when a widget application is blocking), it may appear that the variable is not being updated. The new variable value is displayed when IDL is no longer busy. (Related to ID 49604.) |
| **IDL-66357** | **Installing into a Directory Named with Multibyte  Characters May Cause the IDL Workbench to Not Launch** If you are installing IDL into a directory that contains multibyte characters, path names in the idlde.ini file may be truncated.  **Workaround for 32-bit systems:** In the  <install\_dir>\IDLxx\bin\bin.x86\idlde.ini file, edit the following lines: Line 2 should read: <install\_dir>\IDLxx\bin\bin.x86\jre\bin Line 14 should read: Dosgi.sharedConfiguration.area=<install\_dir>\IDLxx\bin\bin.x86\configuration **Workaround for 64-bit systems:** In the  <install\_dir>\IDLxx\bin\bin.x86\_64\idlde.ini file, edit the following lines: Line 2 should read: <install\_dir>\IDLxx\bin\bin.x86\_64\jre\bin Line 14 should read: Dosgi.sharedConfiguration.area=<install\_dir>\IDLxx\bin\bin.x86\_64\configuration |
| **IDL-66363** | **Executable for a C++ Project that Uses COM\_idl\_connect.tlb Needs to be in \bin directory to Run** If you create a C++ project that uses COM\_idl\_connect.tlb, the .exe must be installed in <install\_dir>\bin\bin.x86 or <install\_dir>\bin\bin.x86\_64 to run. Alternatively, you can add the install path to the PATH environment variable. |

UNIX Platforms

|  |  |
| --- | --- |
| **ID** | **Description** |
| **IDL-14221 IDL-30564** | **DIALOG\_PRINTERSETUP Causes IDL to Hang When Called from a Modal Widget** This is a known problem with no known workarounds except to use DIALOG\_PRINTERSETUP with a non-modal widget. |
| **IDL-41964** | **IDL Does Not Run in Security Enhanced Linux** Some recent versions of Linux implement mandatory access control in the Linux kernel using the Linux Security Modules framework. IDL will not run in such an environment without a modification to the security context for the IDL binaries. **Workaround**: Either disable SELinux or modify the default security context for IDL. See [Help Article 3092](http://www.exelisvis.com/Support/HelpArticleDetail/ArticleId/3092.aspx) for details. |
| **IDL-43301** | **Upper Left Corner of Tree Widget is Never a Valid Drop Zone** When the ability to drag and drop tree widget nodes has been enabled, the UNIX tree widget does not recognize the upper left corner of the widget as a valid drop zone. This area typically extends over the icons of the first two nodes (but not the labels). This is a vendor issue. You can drop items onto the first two nodes by moving the cursor over the labels instead of the icons. |
| **IDL-55427** | **IDL Dataminer Requires latest libstdc++** Third-party libraries used by the IDL Dataminer module require a library that may not be available by default on some newer Linux systems. Attempting to run the IDL Dataminer may cause IDL to issue an error similar to the following:   % DLM\_LOAD: Error loading sharable executable. Symbol: IDL\_Load, File =/rsi/qa/unix/idl71/bin/bin.linux.x86\_64/idl\_dataminer.so libstdc++.so.5: cannot open shared object file: No such file or directory  You will need to install the appropriate libstdc++ compatible library for your platform. |
| **IDL-55537** | **IDLnetURL Objects with ENCODE Property Fails on Linux 32-Bit** Using IDLnetURL with the ENCODE property set to a value other than 0 fails on 32-bit Linux systems. |
| **IDL-57381** | **Missing 64-bit Library** IDL requires a 64-bit version of libXp. Red Hat-based Linux distributions such as RHEL and Fedora may not install this library by default. On Red Hat you can obtain this library using the following command:  yum install libXp.x86\_64 |
| **IDL-58106** | **DIALOG\_PRINTERSETUP Crashes on Linux 64-bit running in the Workbench** Calling DIALOG\_PRINTERSETUP from the Workbench causes 64-bit IDL to crash. **Workaround:**To set up a printer, launch IDL in command line mode and run DIALOG\_PRINTERSETUP. Once you have set up a printer, the information is cached in your home directory. Then printed IDL programs will go to the appropriate printer regardless of whether you are running the Workbench or command line IDL. |
| **IDL-58460** | **IDL Graphics Cause Crash in Ubuntu Versions of UNIX** Recent versions of Ubuntu and other Debian-based Linux distributions may not ship with the libstdc++.so.6 library, which is required by Xerces. You will need to download this library and install it in the /bin/bin.linux.x86 directory of your IDL installation. (On 64-bit installations, this directory is /bin/bin.linux.x86\_64.) |
| **IDL-59497** | **IDL Graphics Fail with Software Rendering on Linux** IDL graphics in the Workbench in software rendering mode may fail on Linux machines if the display is local and Xinerama is not enabled. **Workaround**: Configure Xinerama on the system or run from the IDL command line. |
| **IDL-68960 IDL-68965** | **Java-IDL Bridge with Drawable Windows May Not Render Properly with Software Rendering** You may run into problems when using the Java-IDL Bridge with drawable windows and software rendering. The window may not display anything. Note that this affects the example code in the IDL documentation (IDLWindowExample.java and IDLgrWindowExample.java). **Workaround:** If nothing appears in the drawable window, change to hardware rendering by setting your drawable object's RENDERER property to zero (i.e., RENDERER=0). |

Problems Fixed in IDL 8.3

|  |  |
| --- | --- |
|  |  |
| **IDL-65808** | **TABLE WIDGET ALL\_EVENTS Keyword Change for Windows** On Windows platforms, the ALL\_EVENTS keyword now works as it does in UNIX platforms, returning cell events with keyboard arrow navigation. |
| **IDL-64864** | **On Windows and Solaris, OPeNDAP Functionality Now works with NetCDF** Windows and Solaris users can now take advantage of OPeNDAP functionality when using NetCDF. |
| **IDL-46816** | **Installation Program Failures: Error 1719 now fixed** On some systems, the IDL installation program exited with "Error 1719. Windows Installer service could not be accessed. Contact your support personnel to verify that it is properly registered and enabled." |
| **IDL-62902** | **Licensing issues fixed for Fedora 15, Redhat 6.1, and Later** Updates made to flexlm 11.11.1.1 for IDL 8.3 fixed the issue with licensing. |
| **IDL-68647** | **IDL no longer crashes upon exit or when using graphics.** This was caused by a bug in the glibc libraries. You may need to update your glibc library, or update to RHEL 6.3 or 6.4 will solve the problem. See http://rhn.redhat.com/errata/RHBA-2013-0279.html  for more information. |
| **IDL-66840** | **H5D\_WRITE on IDL 8.1/8.2 MAC 64-bit can now write large datasets.** |
| **IDL-68745** | **Saving to a PDF or EPS file now produces correct lines and text, and no longer clips.** Previously, saving graphics to a PDF file using SAVE caused the output to have an undesirable appearance. |
| **IDL-68883** | **Dataminer installer now correctly populates the ODBC driver registry settings.** Previously you would need to modify the Windows registry manually. |
| **IDL-68399** | **IDL\_IDLBridge and o->EXECUTE /NOWAIT memory leak fixed.** |