**JFace Dialogs : which one is right for you ?**

Posted by **[sureshkrishna](http://www.sureshkrishna.com/)** on March 15, 2008

[**JFace**](http://wiki.eclipse.org/index.php/JFace) framework of [**Eclipse**](http://www.eclipse.org/)provides many standard and useful dialogs and a framework to build custom dialogs and wizards. When the standard dialogs seems to be too simple for your plugin or [**RCP developement**](http://www.eclipse.org/rcp/), one can extend the standard dialogs to suite their own needs. The aim of this article is to provide example oriented approach to dialogs and see in depth of all frequently used dialogs. I am hoping to have this article as the point of reference for many developers to get Dialogs overview.

Dialogs are modal UI components which would do some user interaction tasks. Traditionally Dialogs are used to enter a simple input, to select a choice from a list, to select multiple nodes from a tree, to inform about a success/failure from an operation, to input confirmations from the user, etc…

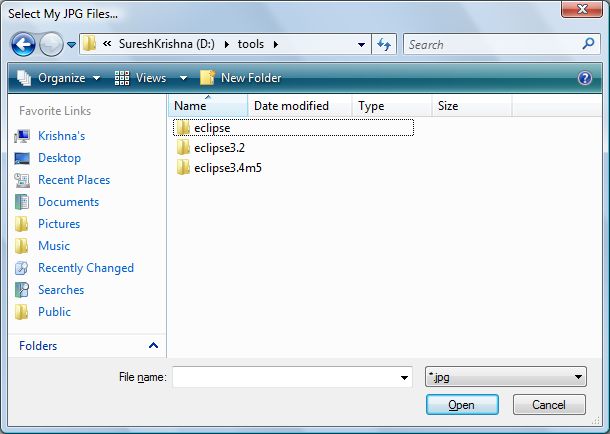
Eclipse provides standard dialogs from two different frameworks, [**SWT**](http://www.eclipse.org/swt/) provides very basic dialogs like *FontDialog*, *ColorDialog*, *DirectoryDialog*, *FileDialog*and*FontDialog*. These [**SWT**](http://www.eclipse.org/swt/)dialogs are coming out of the [**SWT**](http://www.eclipse.org/swt/)as they are very basic and has close connection to the OS. All these Dialogs are available fromorg.eclipse.swt.widgets package from the org.eclipse.swt.<platform>.jar

JFace provides a variety of Dialogs for the implementers to extend or to use them directly. Its interesting to look into variety of dialogs provided by JFace and i would leave the usage to your imagination and context. Rest of this article is mostly code, screen shots and less of textual description. I tried to keep the short, sweet and simple.

**FileDialog:**This is a [**SWT**](http://www.eclipse.org/swt/)Dialog but worth mentioning under the general concept of the Dialogs.

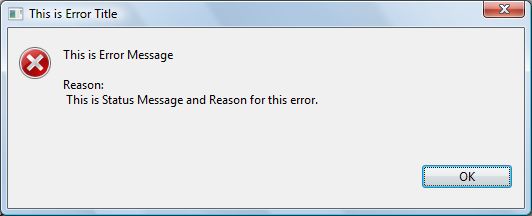
*“Instances of this class allow the user to navigate the file system and select or enter a file name. Note: Only one of the styles SAVE and OPEN may be specified.*

*IMPORTANT: This class is intended to be subclassed <em>only</em> within the SWT implementation.”*

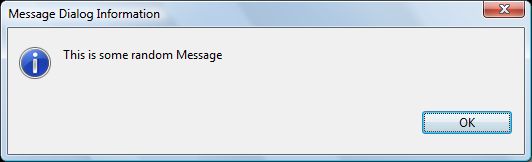
**[](http://sureshkrishna.files.wordpress.com/2008/03/filedialog.jpg)**

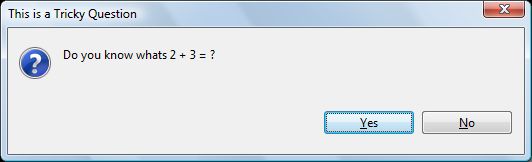
**Information Dialogs:** These dialogs are used to inform user about some event and take some necessary action. There are different Dialogs under this category.

**ErrorDialog :**“*A dialog to display one or more errors to the user, as contained in an IStatus object. If an error contains additional detailed information then a Details button is automatically supplied, which shows or hides an error details viewer when pressed by the user.*“

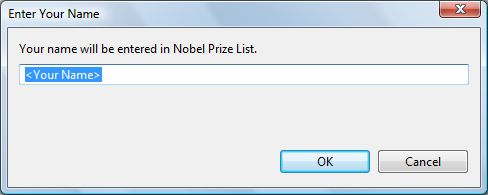
**[](http://sureshkrishna.files.wordpress.com/2008/03/errordialog.jpg)**

**MessageDialog : “***A dialog for showing messages to the user. This concrete dialog class can be instantiated as is, or further subclassed as required.*“

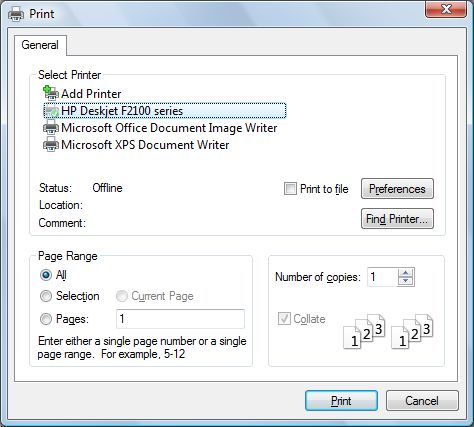
**[](http://sureshkrishna.files.wordpress.com/2008/03/inofrmationdialog.jpg)**

**[](http://sureshkrishna.files.wordpress.com/2008/03/questiondialog.jpg)**

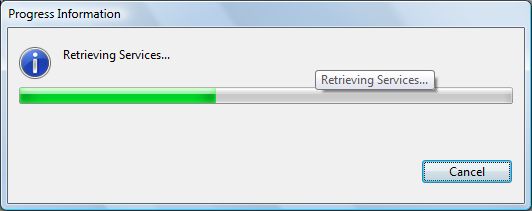
**InputDialog**: This Dialog is used for a very general paradigm to process some input from the user. Generally its not recommended to use this for input of complex data and for large number of input fields.

**[](http://sureshkrishna.files.wordpress.com/2008/03/inputdialog.jpg)**

**PrinterDialog**: “*Instances of this class allow the user to select a printer and various print-related parameters prior to starting a print job. IMPORTANT: This class is intended to be subclassed only within the SWT implementation.*“

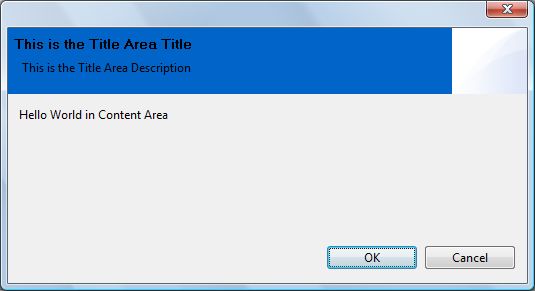
**[](http://sureshkrishna.files.wordpress.com/2008/03/printdialog.jpg)**

**ProgressMonitorDialog**: This is a very important Dialog component most of the applications would use. This Dialog is used to represent any time consuming task or to provide a long running workspace task to the user. Proper care must be taken to show accurate information of progress information to the user.

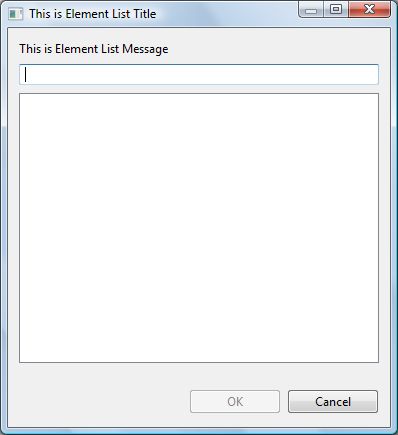
**[](http://sureshkrishna.files.wordpress.com/2008/03/progressdialog.jpg)**

**TitleAreaDialog**: “*A dialog that has a title area for displaying a title and an image as well as a common area for displaying a description, a message, or an error message. This dialog class may be subclassed.*“

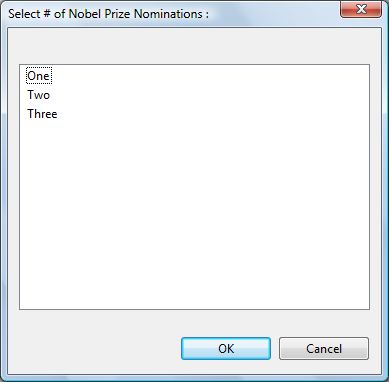
This is the class that i often use for “About MyCompany” kind of dialogs.

**[](http://sureshkrishna.files.wordpress.com/2008/03/titleareadialog.jpg)**

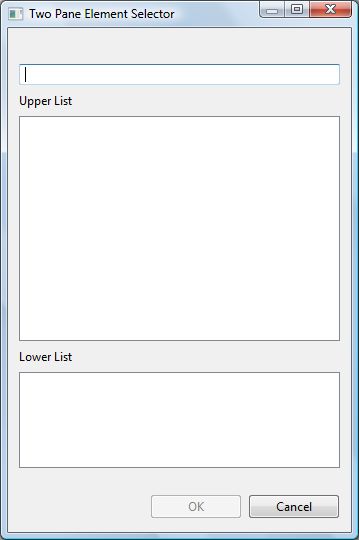
**ElementListSelectionDialog:** “*A class to select elements out of a list of elements.*“

**[](http://sureshkrishna.files.wordpress.com/2008/03/elementlistselectiondialog.jpg)**

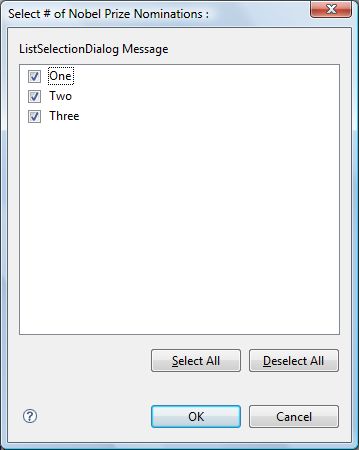
**ListDialog:** “*A dialog that prompts for one element out of a list of elements. Uses IStructuredContentProvider to provide the elements and ILabelProvider to provide their labels.*“

**[](http://sureshkrishna.files.wordpress.com/2008/03/listdialog.jpg)**

**TwoPaneElementSelector:** “*A list selection dialog with two panes. Duplicated entries will be folded together and are displayed in the lower pane (qualifier)*.”

**[](http://sureshkrishna.files.wordpress.com/2008/03/twopanedialog.jpg)**

**List SelectionDialog:** “*A standard dialog which solicits a list of selections from the user. This class is configured with an arbitrary data model represented by content and label provider objects. The getResult method returns the selected elements.*“

**[](http://sureshkrishna.files.wordpress.com/2008/03/listselectiondialog.jpg)**

**ElementTreeSelectionDialog:** “*A dialog class to select elements out of a tree structure.*“

**[](http://sureshkrishna.files.wordpress.com/2008/03/elementtreeselectiondialog.jpg)**

**CheckedTreeSelectionDialog:** “A dialog class to select elements out of a tree structure”

**[](http://sureshkrishna.files.wordpress.com/2008/03/chekcedtreeselectiondialog.jpg)**

**Some Notes :**

* YesNoCancelListSelectionDialog has been Deprecated in 3.3 of Eclipse.
* RCP Product extensions and custmizations can be done via AboutDialogs.
  + **AboutDialog**
  + **AboutFeaturesDialog**
  + **AboutPluginsDialog**
  + **AboutSystemDialog**

Note : Above Dialogs are internal dialogs and not APIs. They are mentioned only as a reference.

* **PrintDialog** is intended to be sub-classes by the clients and use it for identifying the printers and other parameters for print job.
* **SaveAsDialog** is not available in Eclipse 3.3

**Conclusion:** Dialogs are a great way to present simple data and receive user input. Its very important to understand several Dialogs provided by JFace and use them appropriately.

**References:**