DForD SourceCoding User's Manual

DForD Software[1]

October 21, 2010

[1] http://www.dfordsoft.com/

Contents

List of Figures

List of Tables

Chapter 1

Introduce

DForD SourceCoding is a professional development environment and program source code browser. It's familiar and fast and you'll wonder how you ever worked without it.

DForD SourceCoding has variant built-in code analyzers for those popular programming languages including C, C++ and so on. DForD SourceCoding parses all source files in the solution, displays symbols of the currently editing file, especially for C and C++ source code, there is relationship between functions and other symbols which may be displayed. DForD SourceCoding maintains a separated symbol database for each solution, and keeps the database updated, so that users always get the updated symbol and relationship informations.

DForD SourceCoding is designed for large, demanding, real world programming projects.

Chapter 2

License

2.1 DForD SourceCoding SOFTWARE LICENSE AGREEMENT

This user license agreement (the "AGREEMENT") is an agreement between you (individual or single entity) and DForD Software, for the DForD Software software (the "SOFTWARE") that is accompanying this AGREEMENT.

The SOFTWARE is the property of DForD Software and is protected by copyright laws and international copyright treaties. The SOFTWARE is not sold, it is licensed.

2.2 TRIAL VERSION

If you accept the terms and conditions of this AGREEMENT, you have certain rights and obligations as follows:

2.3 YOU MAY

1. Install and use the unlimited number of copies of the TRIAL VERSION of this software for a period of 30 days.
2. Use one or more copies of the TRIAL VERSION for evaluation purposes only.
3. Copy and distribute freely the TRIAL VERSION (see 'DISTRIBUTION').

2.4 YOU MAY NOT

1. Sublicense, rent or lease the TRIAL VERSION or any portion of it.
2. Decompile, disassemble, reverse engineer or modify the TRIAL VERSION or any portion of it.
3. Use the TRIAL VERSION after the trial period of 30 days.

2.5 DISTRIBUTION

You are hereby licensed to make as many copies of the TRIAL VERSION as you wish and distribute it to anyone provided that all files are intact. You must distribute the install form of the TRIAL VERSION (not the files resulting of an installation). You are not allowed to sell the TRIAL VERSION (except for shipment and handling charges). Computer magazines are authorized to distribute it on any Cover Disk or CD-ROM without any permission. Please inform us by e-mail each time you distribute the TRIAL VERSION.

2.6 REGISTERED VERSION

One registered copy of DForD SourceCoding may either be used by a single person who uses the software personally on one or more computers, or installed on a single workstation used non-simultaneously by multiple people, but not both. You may access the registered version of DForD SourceCoding through a network, provided that you have obtained individual licenses for the software covering all workstations that will access the software through the network.

2.7 WARRANTY DISCLAIMER

The SOFTWARE is supplied "AS IS". DForD Software disclaims all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The user must assume the entire risk of using the SOFTWARE.

2.8 DISCLAIMER OF DAMAGES

DForD Software assumes no liability for damages, direct or consequential, which may result from the use of the SOFTWARE, even if DForD Software has been advised of the possibility of such damages. Any liability of the seller will be limited to refund the purchase price.

Chapter 3

Registration

DForD SourceCoding is a "Shareware". You can download and try it for 30 days with no fee. After 30 days, if you decline to register, the software will automatically become disabled. You may register your copy with us for a nominal fee. Finally, every registration enables us to improve our software and continue developing high quality products in the future.

3.1 Price

There are 3 license types for different customers, you can choose the most suitable license to save some spending. Please read the table below to get the detailed license & price information.

Table 3.1: License Type

License Type	**Price**
1 Single License	$149.95 (Buy Now)
Site License for 30 seats	$2249 (Buy Now)
Site License for unlimited seats	$7499 (Buy Now)

For more information, please contact us via sales@dfordsoft.com.

3.2 Registration Benefits

1. Full license to use the software beyond the 30-day evaluation.
2. Remove the Nag window at application startup.
3. Remove the UNREGISTERED mark on the title bar.
4. Free minor upgrades.
5. Discount for major upgrades.
6. Free technical support via e-mail.

Chapter 4

Install/Uninstall

4.1 System Requirement

32-bit Windows XP SP2/SP3 and Windows 2003 are supported. Windows Vista or higher should be worked for, but are not tested.

4.2 Install

Double clicking SourceCodingSetup.exe can install DForD SourceCoding on your computer.

Notice: DForD SourceCoding installation running on Windows XP/2003 does require administrative or "power user" privileges.

4.3 Uninstall

There are 2 ways to remove DForD SourceCoding from your computer.

4.3.1 Method 1

1. **Click the Start Menu on the Microsoft Windows task bar.**
2. **Click Programs.**
3. **Choose DForD SourceCoding within the listed Programs.**
4. **Navigate to DForD SourceCoding uninstall.**
5. **Click Uninstall.**

4.3.2 Method 2

1. **Click Start menu in the Windows task bar.**
2. **Click Settings.**
3. **Click Control Panel.**
4. **Click Add/Remove Program.**
5. **Choose DForD SourceCoding within the "Add/Remove Programs" list of programs.**
6. **Click Add/Remove.**

7. Follow uninstall instructions.

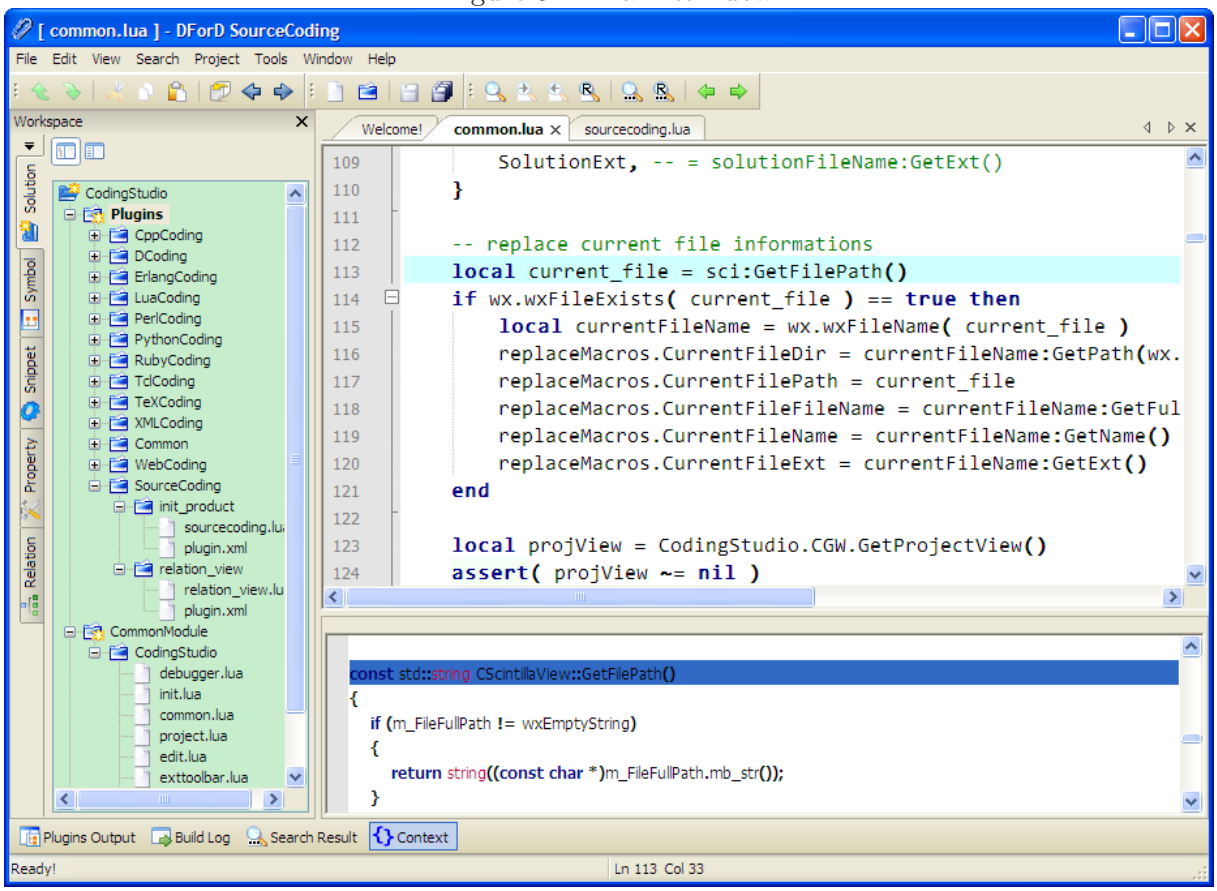
Info: After uninstalling, the Microsoft Windows Uninstall program often does not delete the main folders of programs. You might still see a DForD SourceCoding folder after uninstalling, you may delete the folder manually or just ignore it.

Chapter 5

Features

DForD SourceCoding main window screenshot is shown below.

Figure 5.1: Main Window



5.1 Project Management

DForD SourceCoding provides powerful project management features. As you can see, all project management features can be accessed from the main menu (see Figure 5.2) or the context menu (see Figure 5.3) in Project dockable view.

Figure 5.2: Project Main Menu

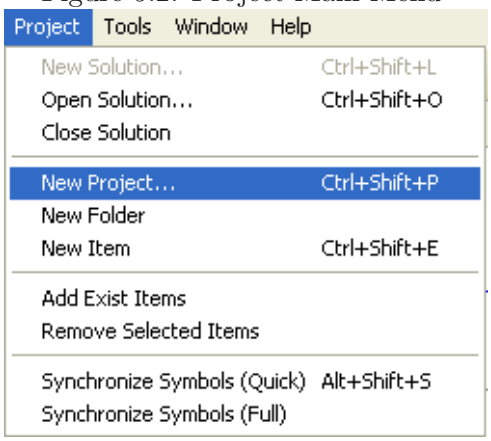
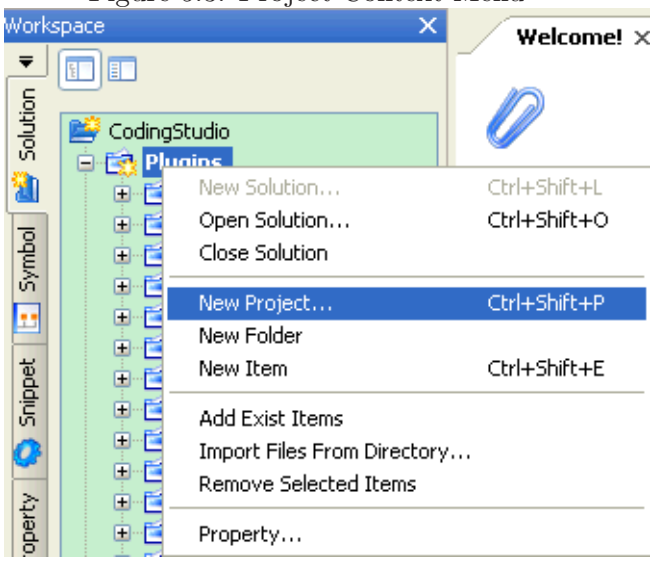


Figure 5.3: Project Context Menu



There are 4 levels for users managing their files, including *solution*, *project*, *folder* and *file*.

DForD SourceCoding can open and manage only one solution at one time. One solution may contain one or more projects. One porject may have one or more folders, and one folder may include one or more files.

Folder is an abstract concept used to classify files, but solution, project and file do represent real files in file system. The solution file takes the extension name *.cssln*, and the project file takes the extension name *.csprj*. Files can be all types of real files except solution files and project files.

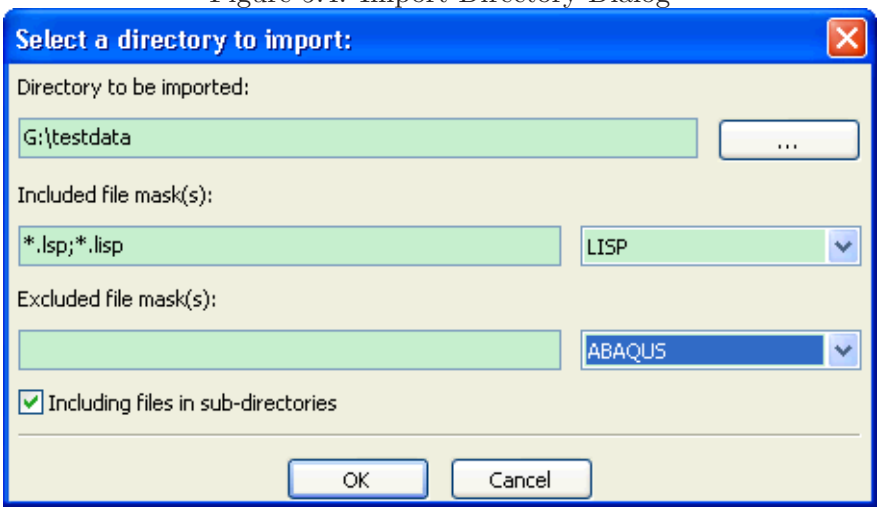
- **New Solution: This command does**
  1. **close the opened solution if exists;**
  2. **ask the user to choose a file path to store the solution file;**
  3. **open this new solution file.**
- **Open Solution: This command does**
  1. **close the opened solution if exists;**
  2. **ask the user to select a solution file from file system;**
  3. **display all projects, folders and files in Project dockable view.**
- **Close Solution: Clear the Project dock-able view and close all opened files belong to this solution.**
- **New Project: Ask the user to choose a file path to store the project file, then add the new project to current opened solution.**
- **New Folder: Create a new folder as the selected project or folder's child node, then the user can rename the new folder.**

- **New Item: Pop up a *New Item* dialog, then the user can choose to create new item which is available in that dialog's candidate list view.**

- **Add Exist Items: Ask the user to choose one or more files, then add the file(*s*) to the selected folder.**

- **Import Files From Directory: This command will popup a dialog (see Figure 5.4) where the user may select and set a directory path, file masks for files to be included or excluded, then the application scans the directory, adds all files and folders retained the same directory structure to the selected project or folder (You can not import a directory to a solution or a file node).**

Figure 5.4: Import Directory Dialog



- **Remove Selected Items: Remove all selected items from Project dock-able view and project file.**

- **Property: Switch to Property dock-able view (see section 5.5), display the selected project item's properties.**

- **Synchronize Symbols(Quick): Traverse all files in the solution, use symbol parsers to extract all symbol information from these files, and store in the solution associated symbol database. This command won't parse those files whose last modified time is older than last parsed time.**

- **Synchronize Symbols(Full): Traverse all files in the solution, use symbol parsers to extract all symbol information from these files, and store in the solution associated symbol database. This command will force to parse all files no matter whether their symbols are updated before, so it would take more time than command Synchronize Symbols(Quick).**

In addition, DForD SourceCoding provides a Quick File Open interface. Users may switch the user interface to the file list by activating accelerator Alt+O or clicking the button on the Project View toolbar, and then input a search keyword so that DForD SourceCoding will filter the file list, only display those ones whose file names contains the search keyword.

5.2 Edit

DForD SourceCoding has an extremely powerful code editor control[1], which supports variant programming languages keyword highlight, source code folding, and so on.

5.2.1 Basic Edit

-  **Undo: Undo one action in the undo history.**
-  **Redo: Redoes the next action on the undo history.**
-  **Cut: Cut the selection to the clipboard.**
-  **Copy: Copy the selection to the clipboard.**
-  **Paste: Paste the contents of the clipboard into the document replacing the selection.**
-  **Select All: Select all the text in the document.**
- **Delete**
  - **Delete Back: Delete the selection or if no selection, the character before the caret.**
  - **Delete Word Left: Delete the word to the left of the caret.**
  - **Delete Word Right: Delete the word to the right of the caret.**
  - **Delete Line Left: Delete back from the current position to the start of the line.**
  - **Delete Line Right: Delete forwards from the current position to the end of the line.**
  - **Delete Line: Delete the line containing the caret.**
- **Format**
  - **Pretty-Print Code Text: Pretty-Print the code text in code editor.**
  - **Make Uppercase: Transform the selection to upper case.**
  - **Make Lowercase: Transform the selection to lower case.**
  - **Increase Line Indent: Increase the selected lines indentation with indent characters which can be set in Configuration dialog General Page(see Section 5.12).**
  - **Decrease Line Indent: Decrease the selected lines indentation with indent characters which can be set in Configuration dialog General Page(see Section 5.12).**
- **Comment**
  - **Block Comment or Uncomment: Comment or uncomment the selected source code lines, editor is able to recognize if it is commented or uncommented.**
  - **Stream Comment: Comment the selected text.**

[1] Scintilla, thanks to Neil Hodgson

- **Insert**
  - **Insert File: Ask the user choose a file, insert this file's content at current caret stayed position.**
  - **Insert Time: Insert current time like *10:23:07* at current caret stayed position.**
  - **Insert Date: Insert current date like *2010-03-16* at current caret stayed position.**
- **Bookmark**
  -  **Toggle Bookmark: Toggle bookmark at current caret stayed line.**
  -  **Previous Bookmark: Goto previous bookmark.**
  -  **Next Bookmark: Goto next bookmark.**
  -  **Clear All Bookmarks: Clear all bookmarks in current activated editor.**

  Notice: Bookmark information is not saved in any forms, so once the user closes the file, all bookmark information is lost.
- **Encoding: DForD SourceCoding supports 4 character encodings, when a file is being opened, DForD SourceCoding checks the byte order mark (BOM[2]) at the beginning of the file, then decides the file's encoding. Also users may modify the editing document's encoding by selecting the specific menu item, DForD SourceCoding will save the document using the specific encoding.**
  - **System Default: Using the system UI default encoding. When the user creates a new file via DForD SourceCoding, this default encoding is always used. If DForD SourceCoding detects a file is neither UTF-8 encoding nor Unicode encoding, then assumes this file is using system UI default encoding.**
  - **UTF-8: Convert current activated document to UTF-8 encoding.**
  - **Unicode Little Endian: Convert currently activated document to unicode little endian encoding.**
  - **Unicode Big Endian: Convert currently activated document to unicode big endian encoding.**
-  **Expand Abbreviation: This command cooperates with Code Snippet (see Section 5.3). The user may input a snippet abbreviation, then activate this command, DForD SourceCoding will expand the snippet at current caret's position.**
-  **Code Snippet Focus Go Back: Move the caret backwards between code snippet completion hot fields.**
-  **Code Snippet Focus Go Forward: Move the caret forwards between code snippet completion hot fields.**

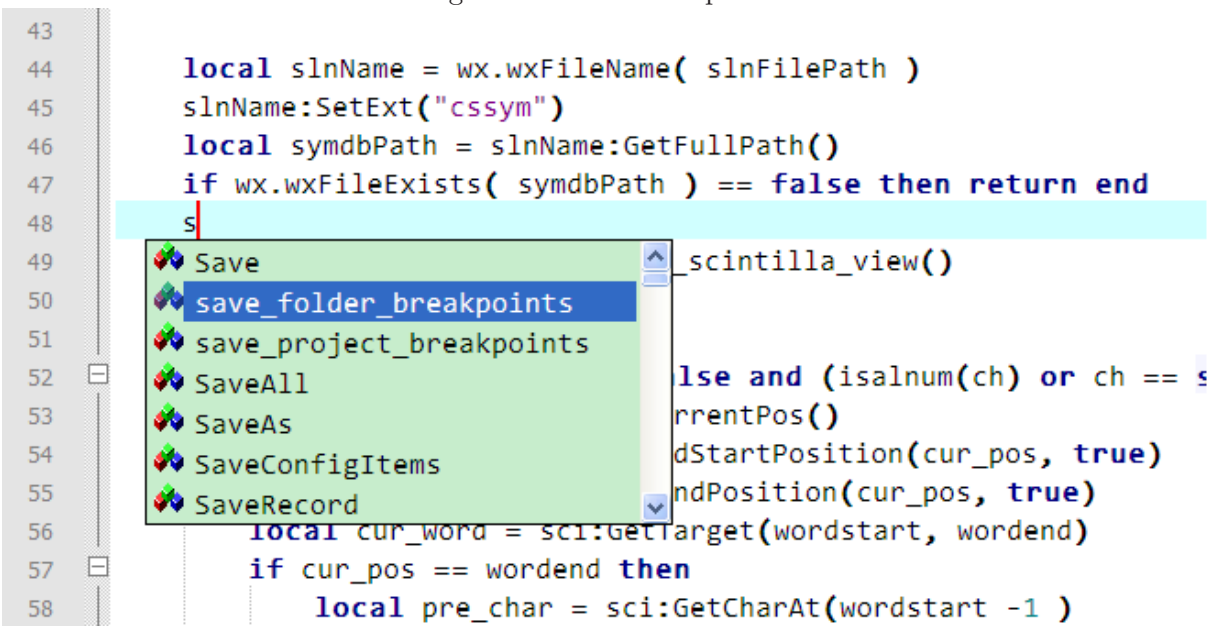
  Tips : We does strongly recommend you using the accelerator (Ctrl+;) to activate this command in order to improve your productivity.

[2] http://en.wikipedia.org/wiki/Byte_order_mark

5.2.2 Auto Completion

Currently DForD SourceCoding supports simple auto-completion (See Figure 5.5), DForD SourceCoding will parse all files in the solution which is opened, and extract all symbols which would be listed in auto-completion candidate list.

Figure 5.5: Auto Completion



Notice: Auto Completion is case-insensitive.

5.3 Code Snippet

DForD SourceCoding provides a set of code snippets for variant programming languages. Especially, DForD SourceCoding's code snippet set for HTML and CSS documents is compatible with Zen Coding. Users can get more detailed informations about Zen Coding from its official project website http://code.google.com/p/zen-coding/.
These snippets can be used in variant types of source files. DForD SourceCoding would detect the current in editing document's type, switch the snippet set and use the right one for the document.
The Snippet View displays all valid snippets for current activated editing document, the user may insert an expanded snippet text by double clicking a list item, or just input the abbreviation then expand it by firing menu item *Expand Abbreviation*.

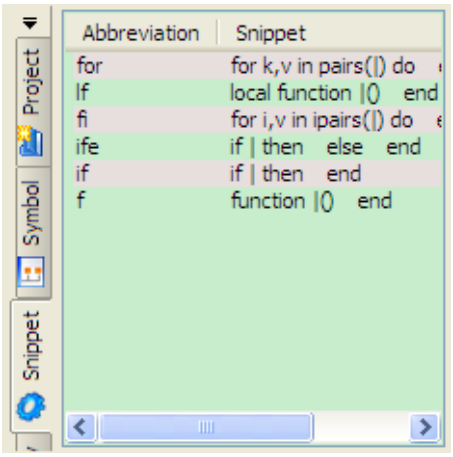


Figure 5.6: Snippet View

We strongly recommend using the second method (the accelerator Ctrl+;) to improve efficiency, and in fact code snippet has been improved, see the example below:

Input text *for* in Lua source document editor, then press the accelerator Ctrl+; DForD SourceCoding will expand the abbreviation *for* into *for i = 1, 10 do ...*, and select the loop variable *i*. The user may modify the loop variable name, or

just press the accelerator Ctrl+. to jump to selecting the loop starting value *1* and press the accelerator Ctrl+. again to jump to selecting the loop ending value *10*.

Please remember that the accelerator Ctrl+. is used to selecting the right hot field, and the accelerator Ctrl+, is used to selecting the left hot field.

For more usage tips, please watch the video online at http://www.dfordsoft.com/cs/lua_video.htm**.**

5.4 Symbol View

DForD SourceCoding analyzes current editing source code, extracts all parser-defined symbols and list them in Symbol View. As you can see, most popular programming languages are supported, and users may customize the parsers in Configuration dialog(see section 5.9).
The user may type any text in the search text control, DForD SourceCoding will filter the symbols, and list those symbols which contain the user input text only. This helps users find and locate the symbols quickly.
If the user double clicks an item in the Symbol view, DForD SourceCoding will jump to the line where that function is defined.

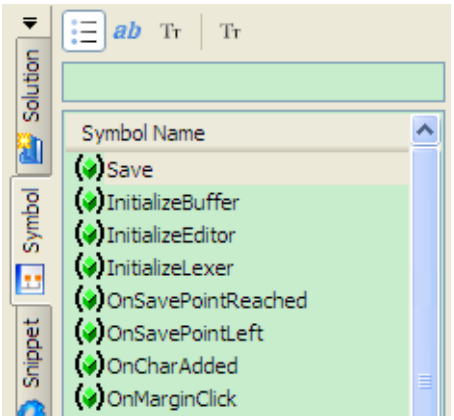


Figure 5.7: Symbol View

5.5 Property View

The Property View displays several properties of selected item in Project view.
If there are more than one items being selected, only the first one's properties are displayed.
As we has descripted in Section 5.1 Project Management, there are 4 types of project items: solution, project, folder and file. Each type's properties displayed in Property View are listed in the table below.

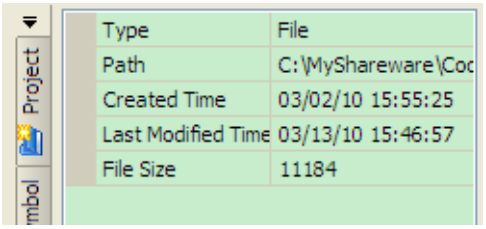


Figure 5.8: Property View

Table 5.1: Properties For Project Items

Item Type	Propertis
Solution	Type, Solution file path
Project	Type, Project file path
Folder	Type, Folder name
File	Type, File path, File created time, File last modified time, File size

5.6 Relation View

DForD SourceCoding may display symbols' relationship in Relation View. It is supported by an excellent application called cscope(For more information about cscope please visit its official website http://cscope.sourceforge.net/). Because of the cscope application's limitation, Relation View supports C/C++ files only.
Relation View provides 4 types of relationship informations: functions called by this function, functions call this function, code lines which contains this C symbol, code line which defines this C symbol. But DForD SourceCoding can only display one type of these relationship informations at one time. Users may switch the displaying type by clicking the tool button on Relation View, Relation View would refresh the tree view immediately.
All these relationship information are displayed as a tree, the higher level nodes represent the source code file, and the lower level leaf nodes represent the source code line. If users double click a leaf node, DForD SourceCoding will open that source file, locate the caret at the beginning of that line.

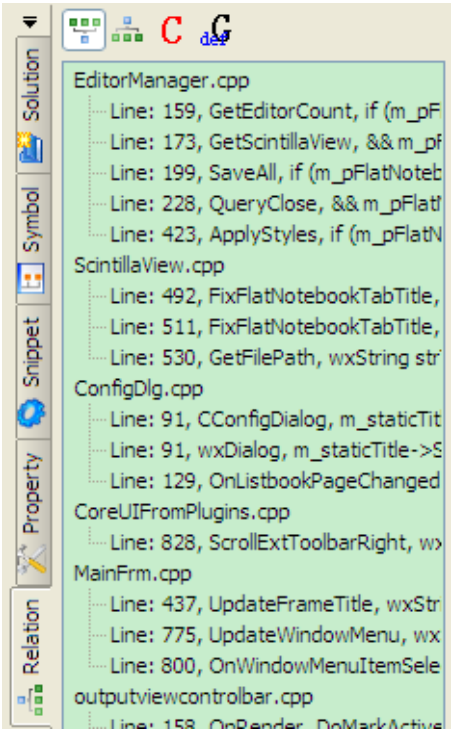
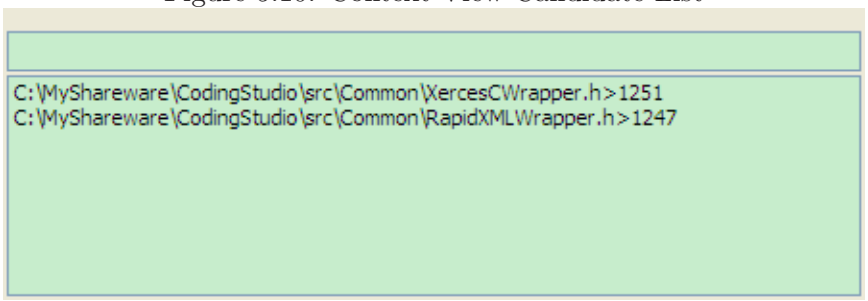


Figure 5.9: Relation View

5.7 Context View

DForD SourceCoding may try to find the symbol's definition position, and display that source code in Context View. DForD SourceCoding keeps the symbol database updated, so that Context View almost always displays the right source code. And users may modify the programming languages' symbol parsers in Configuration dialog (see section 5.9), in order to get more accurate informations.

Figure 5.10: Context View Candidate List



5.8 Find & Replace

DForD SourceCoding has basic search and replace feature.

- **Find: Find text starting.**

  Notice: Search/Replace in regular expression may be slower than normal search. Currently the regular expression support is limited and should only be used for simple cases and initial development. Integrating a Perl-Compatible

regular expression engine is working in progress. In a regular expression, special characters interpreted are list in Table 5.2.

- Find Next: Find text starting at current archor.

  Notice: If you have never searched for any string before, DForD SourceCoding won't find anything.

- Find Previous: Find some text starting at the search anchor and moving backwards.

  Notice: If you have never searched for any string before, DForD SourceCoding won't find anything.

- Replace: Replace the target text with the argument text.

- Find In Files: Find text starting in files.

  The user can search text in current activated document, or current selected project, or current opened solution, or even any real directory.

  The user can specify searched file type whose syntax is compatible with DOS file system wildcard character rule. If there are more than one types, file type patterns should be separated by semicolon.

  All matched items will be displayed in Search Result list view(see Figure 5.11), the user may double click the list item, DForD SourceCoding will open that file and jump to the specific line.

- Replace In Files: Replace the target text with the argument text in files.

  Notice: Search/Replace In Files can use Perl-Compatible Regular Expression engine.

- Go Back: Move caret to the previous position.

- Go Forward: Move caret to the next position.

Figure 5.11: Search Result View

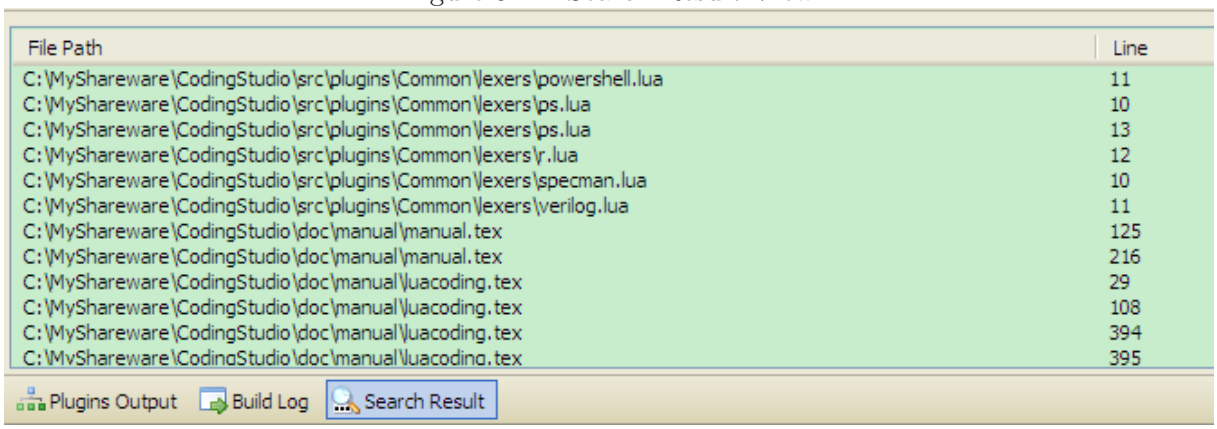


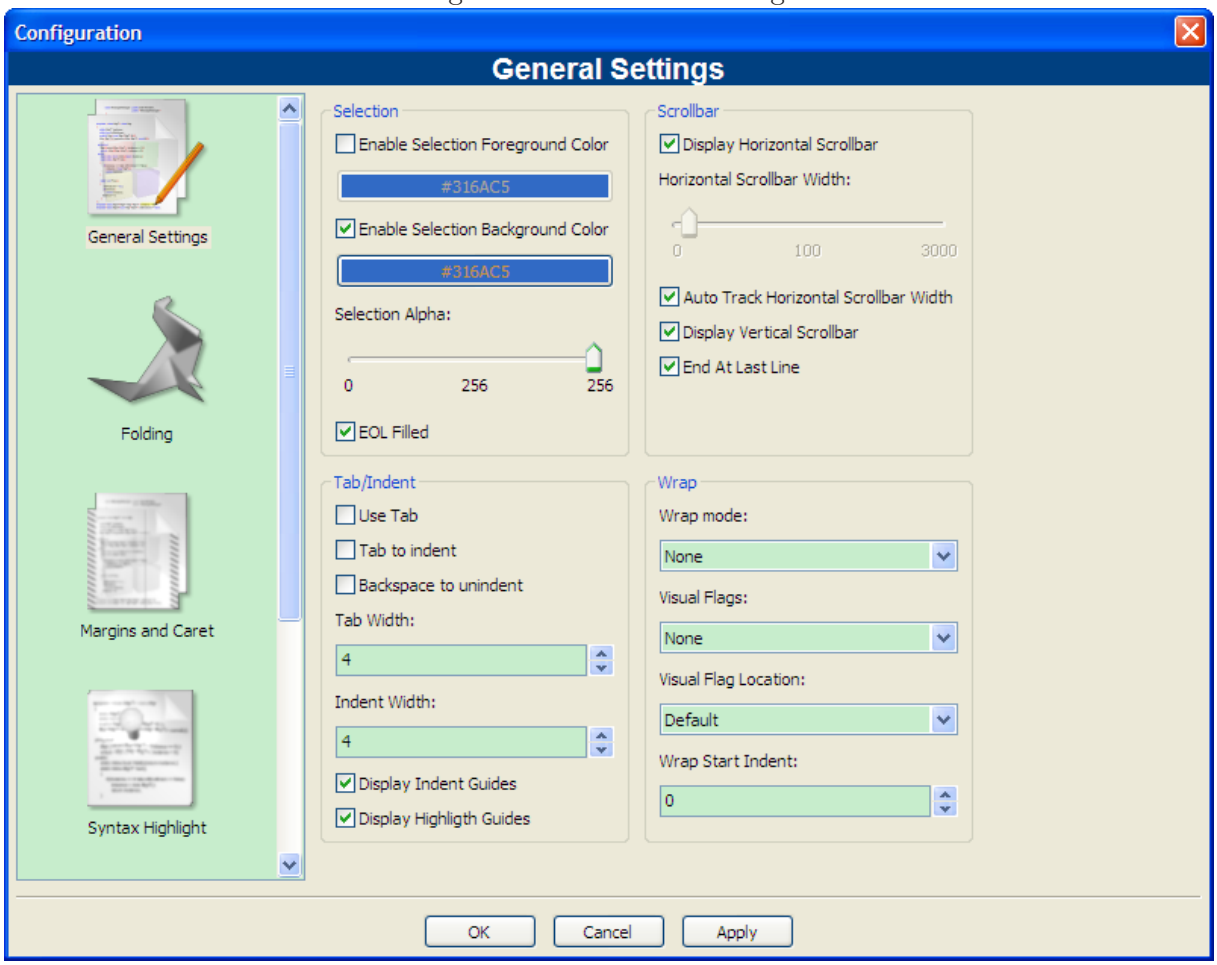
Table 5.2: Search In Regular Expression Special Characters Interpret

Characters	**Interpreting**
.	Matches any character
\(	This marks the start of a region for tagging a match.
\)	This marks the end of a tagged region.
\n	Where n is 1 through 9 refers to the first through ninth tagged region when replacing. For example, if the search string was **Fred\([1-9]\)XXX** and the replace string was **Sam\1YYY**, when applied to **Fred2XXX** this would generate **Sam2YYY**.
\<	This matches the start of a word using editing programming language's definitions of words.
\>	This matches the end of a word using editing programming language's definition of words.
\x	This allows you to use a character x that would otherwise have a special meaning. For example, [would be interpreted as [and not as the start of a character set.
]...]	This indicates a set of characters, for example, [abc] means any of the characters a, b or c. You can also use ranges, for example [a-z] for any lower case character.
]^...]	The complement of the characters in the set. For example, [^A-Za-z] means any character except an alphabetic character.
^	This matches the start of a line (unless used inside a set, see above).
$	This matches the end of a line.
*	This matches 0 or more times. For example, Sa*m matches Sm, Sam, Saam, Saaam and so on.
+	This matches 1 or more times. For example, Sa+m matches Sam, Saam, Saaam and so on.

5.9 Configuration

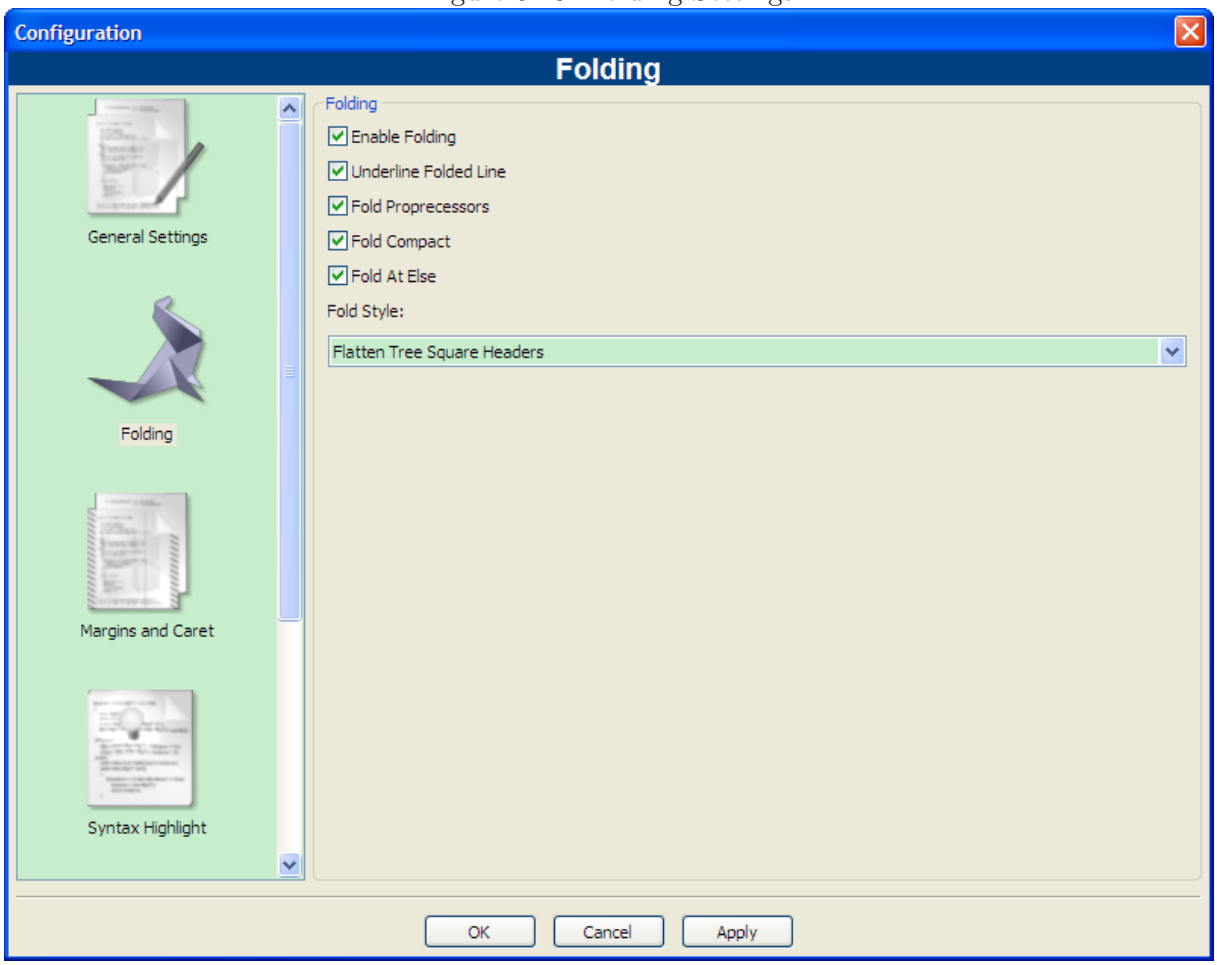
Users may customize application settings in Configuration dialog. All options are divided into 6 categories, including General Settings(see Figure 5.12), Folding(see Figure 5.13), Margins and Caret(see Figure 5.14), Syntax Highlight(see Figure 5.15), Symbol Parser(see Figure 5.16) and Keyboard Shortcut(see Figure 5.17). Each category is placed in a separated configuration page.

Figure 5.12: General Settings



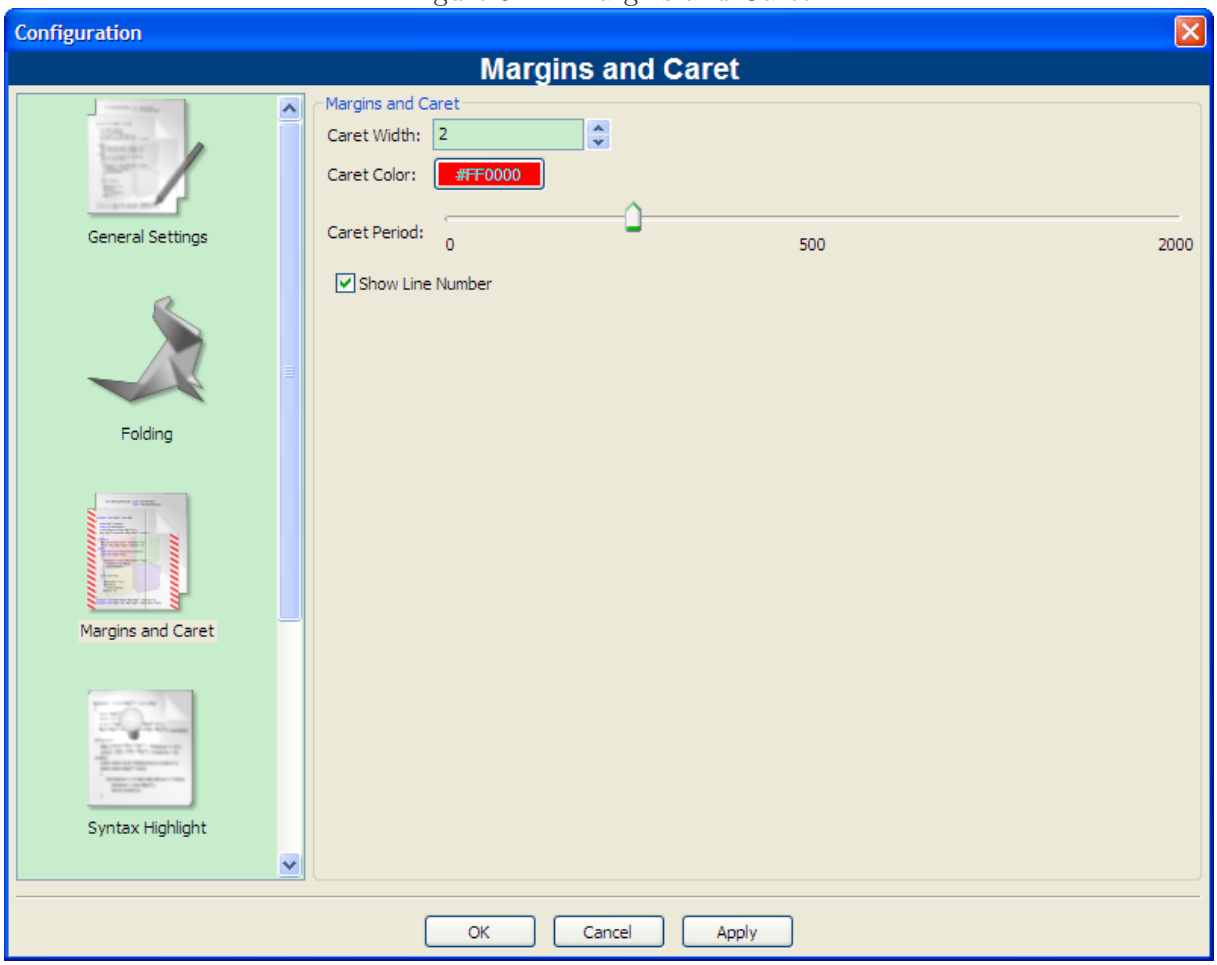
There are many options in General Settings config page. Users may set the selection's colors, scroll bars' visibility, tab and white space, and text line wrap options.

Figure 5.13: Folding Settings



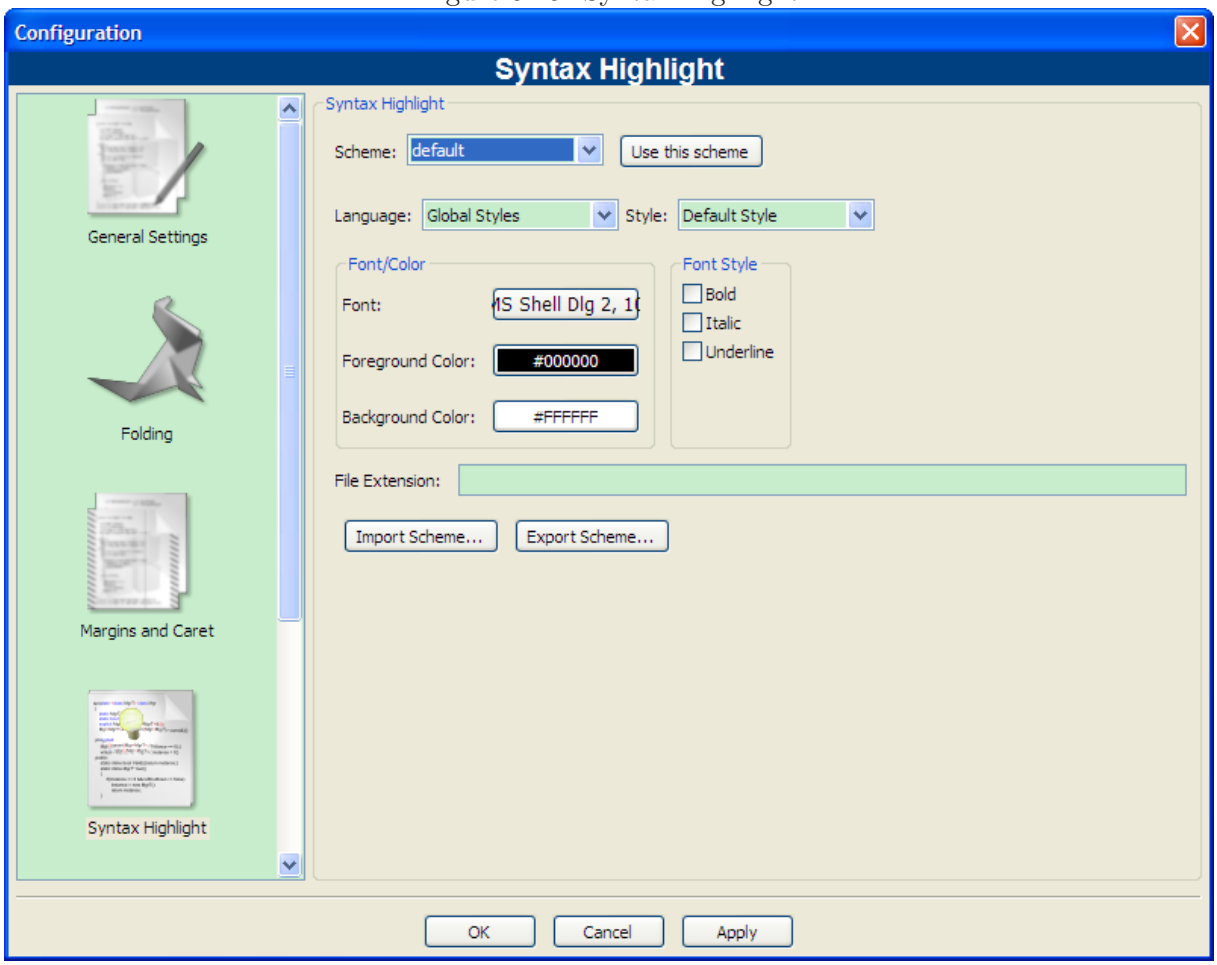
In this Folding config page, users may set to show or hide folding margins in code editors, and styles of folding symbols. There are 4 types of styles for folding symbols, *Flatten Tree Square Headers* style is used as default style.

Figure 5.14: Margins and Caret



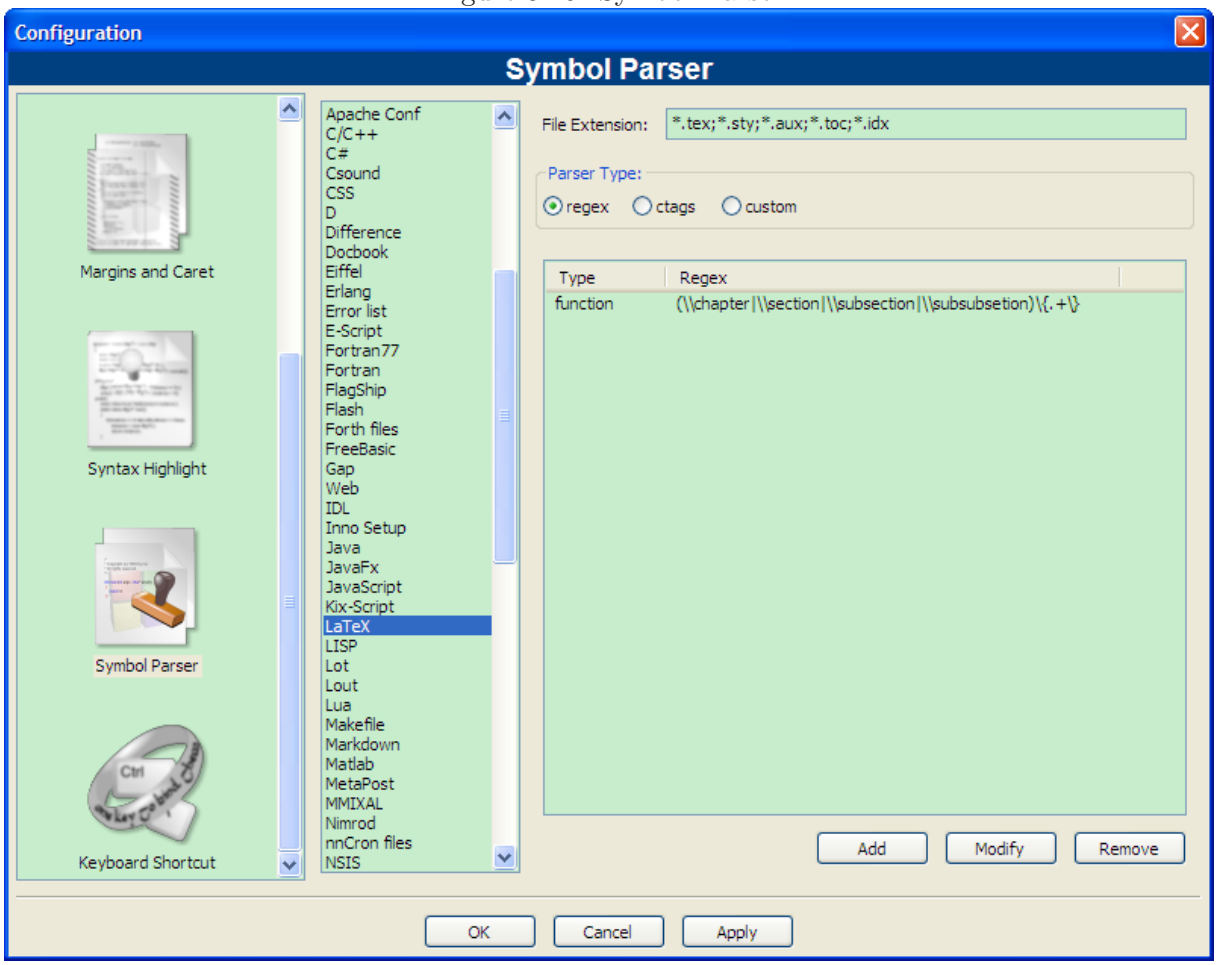
In this Margins and Caret config page, users may set the caret width, color and blinking rate. Also users may choose to show or hide line number margin in code editors.

Figure 5.15: Syntax Highlight



DForD SourceCoding provides syntax highlight for more than 80 programming languages. As you can see from figure 5.15, in Syntax Highlight config page, users may customize syntax highlight details including font name, size, background/foreground colors, font styles and so on. Users may exports the syntax highlight settings into a file which is called Scheme by DForD SourceCoding. So that users may share these settings with other people.

Figure 5.16: Symbol Parser



Symbol Parsers are used to parse source code and extract syntax symbols which help users understanding the source code. These extracted syntax symbols are used in several ways. Symbol View(see Section 5.4) displays symbols from symbol parser, Context View displays symbol definition information from symbol database which is maintain by symbol parser, etc. DForD SourceCoding has several built-in symbol parsers which support some popular programming languages such as C, C++, Java, Lua, LaTeX and so on.

There are 3 types of symbol parsers: regex, ctags and custom.

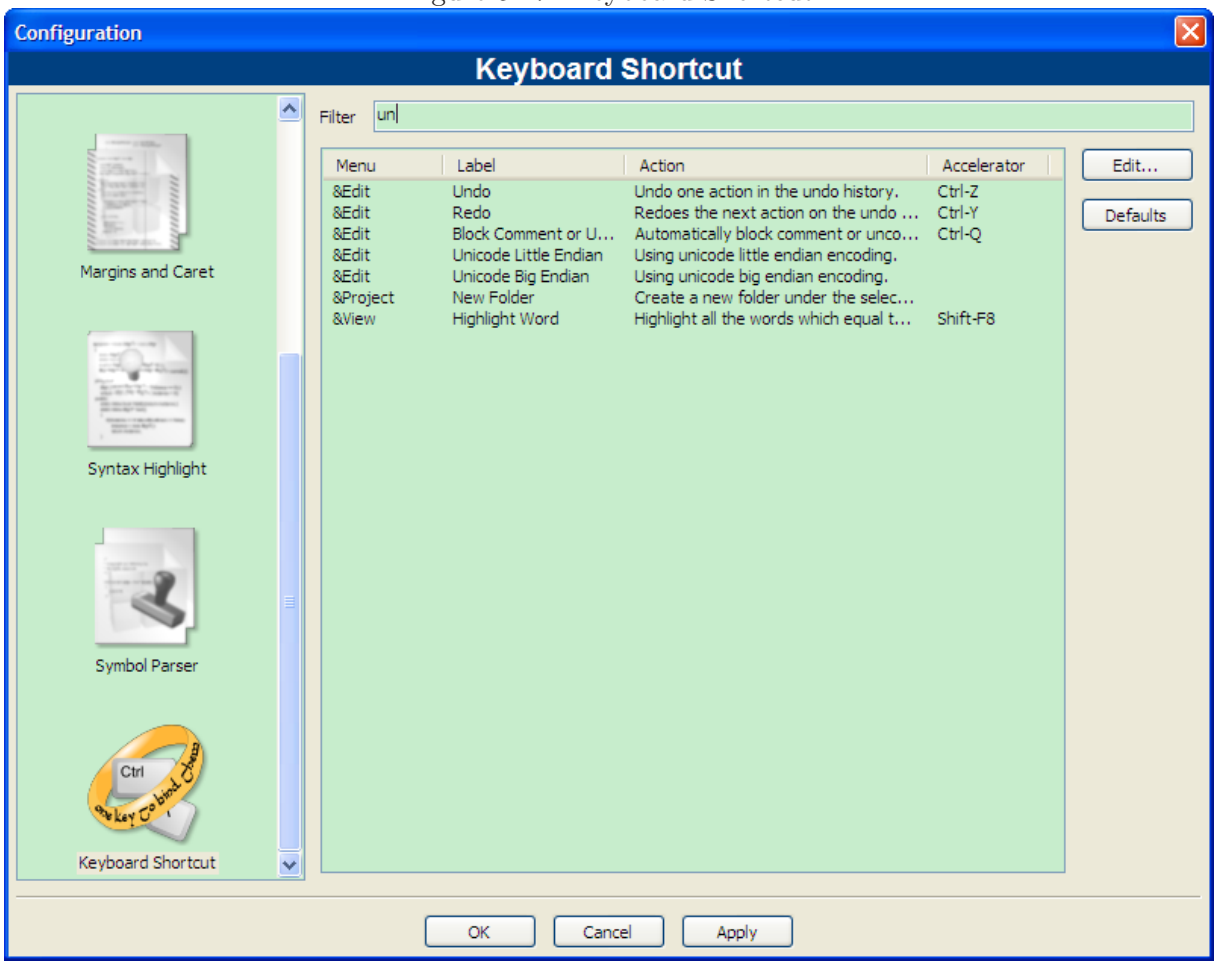
1. **Users may define some regular expressions which are used by DForD SourceCoding to match the full source code text and extract symbols.**

2. **Ctags[3] is a powerful program which could extract tags from more than 40 programming language source code, DForD SourceCoding reads ctags's output tags as the symbol parsers' results.**

3. **Custom parser means users may provide much more accurate syntax parsers for a specific programming language (eg. A special Lua parser component is shipped with DForD SourceCoding.), so that more detailed informations would be extracted.**

[3] http://ctags.sourceforge.net/

Figure 5.17: Keyboard Shortcut



In Keyboard Shortcut config page, users may modify any menu item binding command's keyboard shortcut. The modified settings would be loaded at application startup.

Advance users always use accelerators to improve their working effectivity, this feature may help users greatly. Users may input search keyword to filter commands by action field. Also application provides a default settings recovery feature, so that any one could recover the application's keyboard shortcut settings at any time.

5.10 External Tools

Users may add commands as external tools by selecting menu item Tool - External Tools, a External Tools dialog should popup(see Figure 5.10), users may set external tool's title, command, arguments and so on. Every external tool is displayed as a menu item under Tool menu, so that users may run the command without leaving DForD SourceCoding.
There are several macros for arguments and initial directory, all of these macros are used to represent a filesystem path, the tables below shows more detailed description about these macros. Users may input macros by selecting popup menu item on clicking the right button quickly.

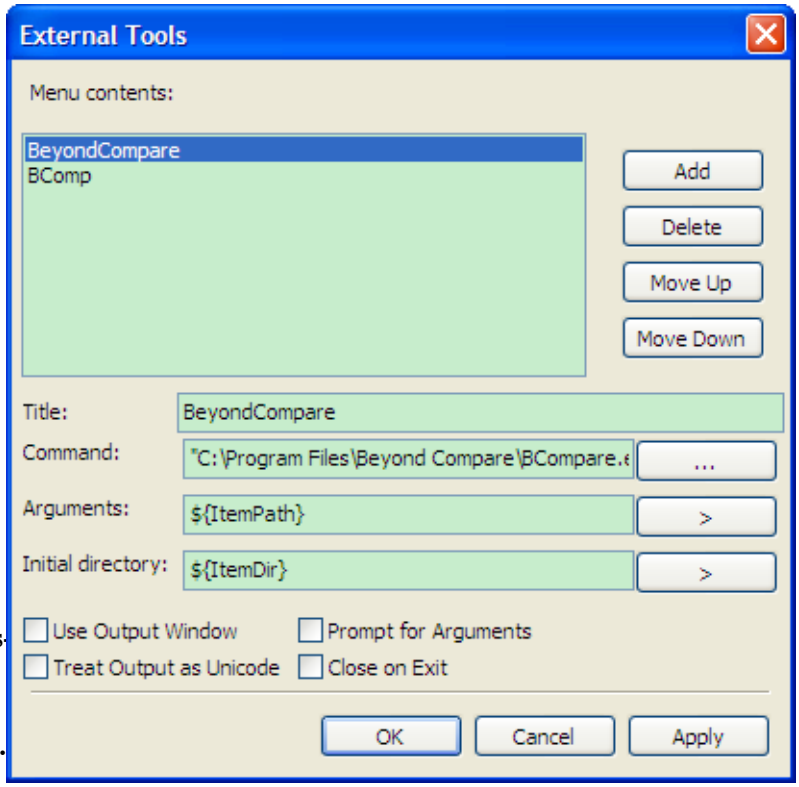


Figure 5.18: External Tools

Table 5.3: Macros for Arguments

Macro	Description
Solution	Type, Solution file path
Project	Type, Project file path
Folder	Type, Folder name
File	Type, File path, File created time, File last modified time, File size

Table 5.4: Macros for Initial Directory

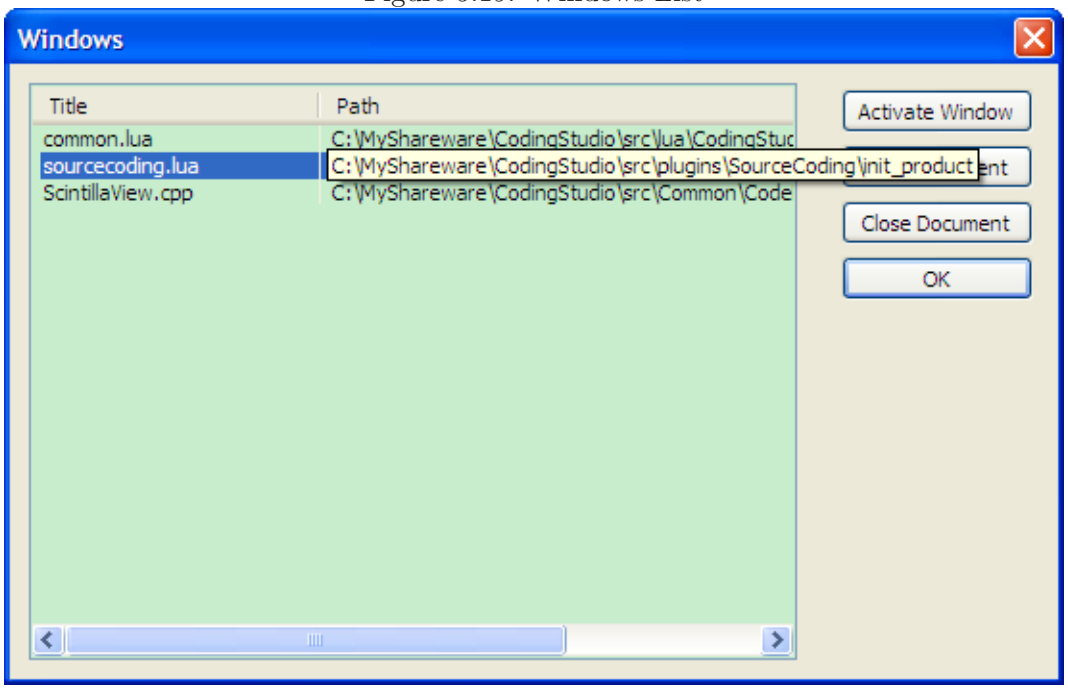
Macro	Description
Solution	Type, Solution file path
Project	Type, Project file path
Folder	Type, Folder name
File	Type, File path, File created time, File last modified time, File size

5.11 Windows List

DForD SourceCoding supports multi-document user interface, users may open several documents and each document takes a tab. But as you can see, the foreground frame may not display all tabs if the user opens many documents, it is not very convenience to locate the specific tab directly. DForD SourceCoding provides Windows List to resolve this issue. Users may open the Windows List by selecting menu item Window - Windows..., then a Windows dialog should popup, and all opened

documents' titles and filesystem paths are listed in the list control(see Figure 5.19).

Figure 5.19: Windows List



Chapter 6

Frequently Asked Questions

- **Q: How to buy DForD SourceCoding?**

  A: You can buy it online at http://www.dfordsoft.com/cs/sc_buy.htm.

- **Q: How long will it take to receive my serial number?**

  A: After placing an order, you will be able to receive your serial numbers immediately from RegNow. If you fail to receive your serial number after 24 hours, please contact support@dfordsoft.com.

- **Q: I don't like some of the text colors, how to change it?**

  A: Select menu item Tool - Options, goto Syntax Highlight page, you can customize every syntax element for every supported programming language as you like.

- **Q: Where does DForD SourceCoding save configuration files and temporary files?**

  A:

  1. **DForD SourceCoding configuration files will be saved in: C:\Documents and Settings\<User>\Application Data \CodingStudio, we strongly recommend you do not modify or delete any of these files, otherwise DForD SourceCoding may not work correctly.**
  2. **DForD SourceCoding temporary files will be saved in: C:\Documents and Settings \<User>\Local Settings \Temp. DForD SourceCoding will remove the temporary files by default.**

- **Q: I need function xxx, what can I do in DForD SourceCoding?**

  A: OK, if the function you need does not exist in DForD SourceCoding, you can tell us, we will eliminate the requirement, and may implement it in later versions.

Chapter 7

Feedback and Support

7.1 Homepage

If you want to get the latest information about DForD SourceCoding or any other DForD softwares, please visit DForD Software Homepage(http://www.dfordsoft.com/).

7.2 Feedback

Please contact us via e-mail feedback@dfordsoft.com if you find a bug or have any suggestions.

7.3 Get your lost registration code

You can mail to support@dfordsoft.com. Please tell us your registration name, E-mail address and DForD SourceCoding version number. We will check your registration information and mail your the registration code.

7.4 Support

If you need help, or for any other reasons, you can mail to support@dfordsoft.com, we provide free technical support via e-mail for registered users.

Chapter 8

Acknowledgments

DForD SourceCoding uses a lot of third party libraries (see Table 8.1), we would like to express our appreciations to all of these authors for making the libraries available.

Table 8.1: Third Party Libraries

Library Name	**Author**	**Homepage**
wxWidgets	The wxWidgets Team	http://www.wxwidgets.org/
Boost	The Boost Team	http://www.boost.org/
Lua	The Lua Team	http://www.lua.org/
Luabind	Daniel Wallin and Arvid Norberg	http://www.rasterbar.com/products/luabind.html
wxLua	The wxLua Team	http://wxlua.sourceforge.net/
Scintilla	Neil Hodgson	http://www.scintilla.org/
wxScintilla	Otto Wyss	http://wxcode.sourceforge.net/components/wxscintilla/
Sqlite3	Richard Hipp	http://www.sqlite.org/
wxFlatNotebook	Eran Ifrah	http://wxflatnotebook.sourceforge.net/
wxPropertyGrid	Jaakko Salli	http://wxpropgrid.sourceforge.net/
PCRE	Exim MTA	http://www.pcre.org/
lrexlib	Reuben Thomas and Shmuel Zeigerman	http://math2.org/luasearch/rex.html
LuaSQL	Pedro Miller Rabinovitch and Roberto Ierusalimschy, Kepler Team	http://www.keplerproject.org/luasql/
LuaBitOp	Mike Pall	http://bitop.luajit.org/index.html
lua-ex	Mark Edgar and Rici Lake	http://luaforge.net/projects/lua-ex/
Win32 API wrappers	Daniel Quintela	http://luaforge.net/projects/w32wrappers/
zip	Andre Carregal, Luis Eduardo Jason Santos	http://luaforge.net/projects/luazip/
lpeg	PUC-Rio	http://www.inf.puc-rio.br/~roberto/lpeg/lpeg.html
ICU	IBM	http://site.icu-project.org/
ICU4Lua	Duncan Cross	http://luaforge.net/projects/icu-lua/
RapidXML	Marcin Kalicinski	http://rapidxml.sourceforge.net/