

Getting Started in Crystal C/C++

1. Crystal C Menu
2. Online Help in Crystal C
3. View Object's Type
4. Function Outline
5. Globals, Functions etc. from each Source File
6. Move Comments to the Right
7. Find in Files (GREP) & Search in Current File
8. Navigating with Quick Buttons

Crystal C Menu

The screenshot displays the Crystal REVS for C IDE interface. The main window shows the source code for `Jsscope.c` at Row 258, Col 48, Lines 515. The code includes functions like `js_list_scope_lookup` and `js_list_scope_add`, along with a `#define HASH_THRESHOLD 5` and various data structures like `JSSymbol`, `JSScopePrivate`, and `JSHashTable`.

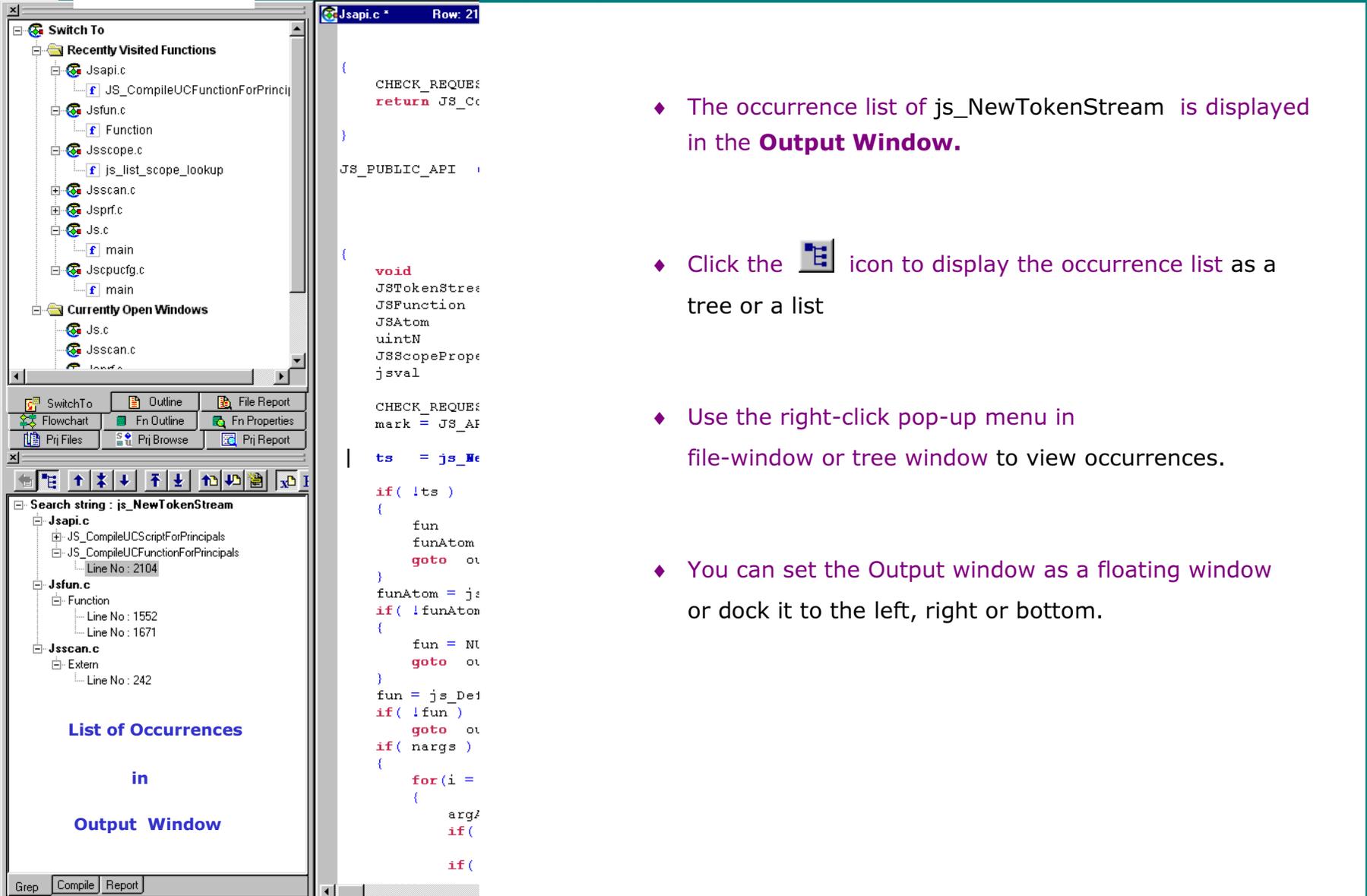
On the left side, there is a **Browse Window** showing a tree view of the project structure under `Switch To` and `Currently Open Windows`. The `Currently Open Windows` section lists `Jsscope.c`.

At the bottom left, there is a toolbar with buttons for `SwitchTo`, `Outline`, `File Report`, `Pri Files`, `Pri Browse`, `Pri Report`, `Flowchart`, `Fn Outline`, and `Fn Properties`.

On the right side, there is a **Quick Buttons** panel with various navigation and editing buttons, including `Cd&Cmt`, `UDD`, `AnyErr`, `Unusd`, `FileStt`, `BookMark`, `FileEnd`, `Line#`, `Home`, `End`, `ScrTop`, `ShwDefn`, `ScrMid`, `ShwDecl`, `ScrBot`, `PopCnbt`, `MatchPair`, `DelPair`, `Page`, `GoFwd`, `Page`, `GoBack`, `Skip`, `Fun`, `Skip`, `Fun`, `Outer`, `Marker`, `RptSrch`, `Marker`, `InsLn`, `Srch`, `InsLn`, `Srch`, `Sntnc`, `Obj`, `Sntnc`, `Obj`, `Desel`, `WrkBack`, `Mdfd`, `DelLine`, `Mdfd`, `DelWord`.

At the bottom right, there is a **Tabs** panel with buttons for `Edit`, `Comment`, `TALL`, `Wide`, `Full Screen`, and `Help`.

List of Occurrences



The screenshot displays a development environment with the following components:

- Browse Window:** Shows a project tree with folders like 'Recently Visited Functions' and 'Currently Open Windows'. The 'Jsapi.c' file is selected.
- Jsapi.c Editor:** Shows C code with a search string 'js_NewTokenStream' applied. The code includes function definitions and calls to 'js_NewTokenStream'.
- Output Window:** Displays the 'List of Occurrences' for 'js_NewTokenStream' in a tree view. The occurrences are listed with their file names and line numbers:
 - Jsapi.c: JS_CompileUCScriptForPrincipals (Line No: 2104)
 - Jsfun.c: Function (Line No: 1552, Line No: 1671)
 - Jsscan.c: Extern (Line No: 242)

Annotations on the screenshot:

- List of Occurrences** (in blue text)
- in** (in blue text)
- Output Window** (in blue text)

Buttons at the bottom of the Output Window: Grep, Compile, Report.

- ◆ The occurrence list of `js_NewTokenStream` is displayed in the **Output Window**.
- ◆ Click the  icon to display the occurrence list as a tree or a list
- ◆ Use the right-click pop-up menu in file-window or tree window to view occurrences.
- ◆ You can set the Output window as a floating window or dock it to the left, right or bottom.

Online Help in Crystal C/C++

1. Online documents:

- Click [Start](#) → [Programs](#) → [Crystal C and C++](#) → [Online Documents](#)

2. Right-click on any button or tab for help on that button or tab.

- Similarly, **right click** on any command [in the pull-down menus](#).

3. Current Menu Help: click the **<Help> tab** (at the bottom of the window).

[Help-topic-tabs](#) are displayed when you are editing with the Tokens Panel or the Comments panel, or you are in the Flowchart Menu or Tree menu.

Click each [help-topic-tab](#) to get an overview of the current menu.

To remove the [Help-topic-tabs](#), [deselect the <Help> tab](#).

- [In menus](#) such as the Open-File menu, click the **<Help>** button in that menu.

4. Use the **Help** Pull-down Menu.

Overview

After starting Crystal C, you can:

- ◆ Work on files as stand-alone files:
Set Options, Specify Include-Path-Sequence, then open the files.
- ◆ or Create a Project, Set Options, Add files, Specify Include-Path-Sequence, then open the files.

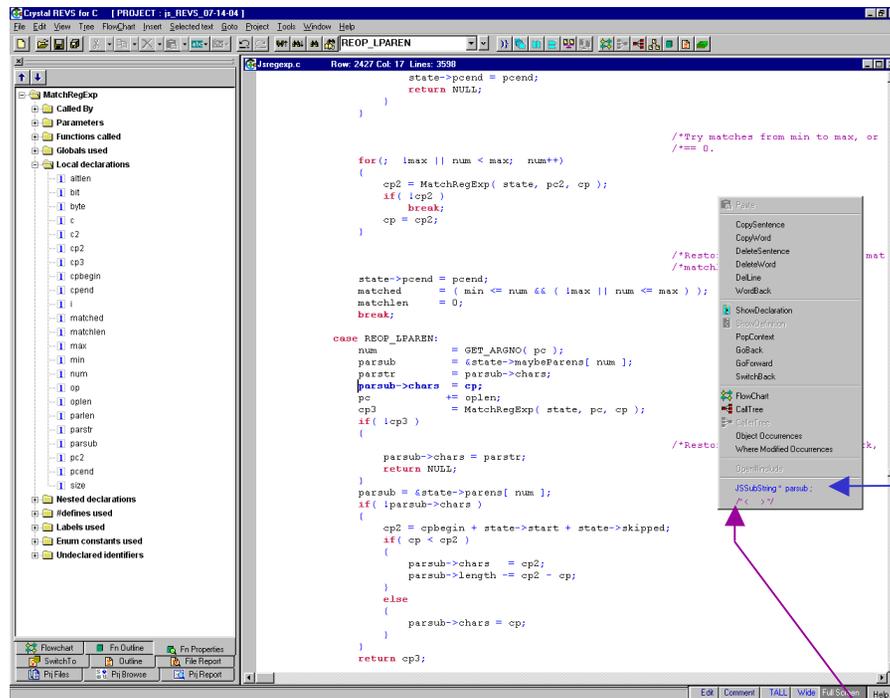
(Please refer to Chapter 3 "Set up a Project and Open files".)

After opening one or more files,

Click the  icon in the toolbar to go to any function in the file; use Quick buttons to navigate.

1. Use the Tokens Panel for editing – click the  tab.
2. Create the flowchart of the current function – click the  icon in the toolbar.
3. Create the Call-tree of the current function – click the  icon in the toolbar.
4. Use the Comments Panel for editing – click the  tab.

View Object's Type



(The cursor is on the object-name `parsub`.)

Right-click anywhere in the window.

The right-click pop-up menu is displayed.

- Type-information of the object.
Click on it to view full type-information.

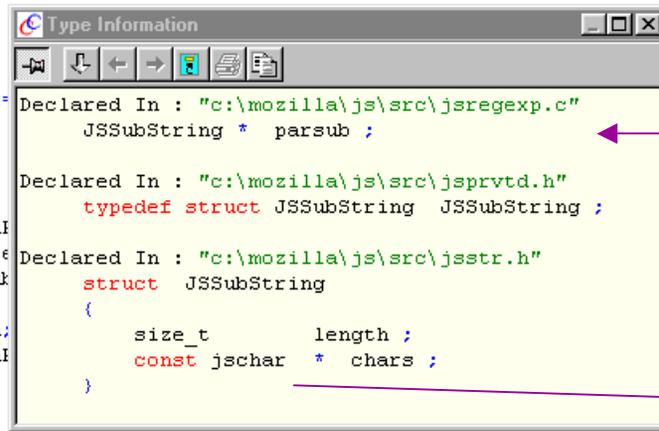
- Meaning of the object

Full Type-Information

```
    if( !cp2 )
        break;
    cp = cp2;
}

state->pcend = pcend;
matched     = ( min <=
matchlen    = 0;
break;

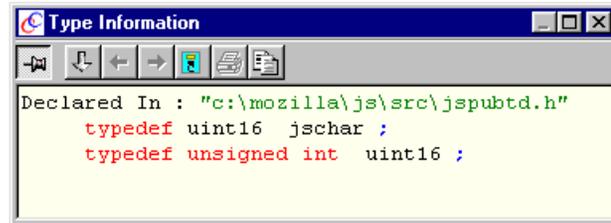
case REOP_LPAREN:
    num      = GET_ARG;
    parsub   = &state->parstr;
    parstr   = parsub->chars;
    pc       += olen;
    cp3      = MatchLen;
    if( !cp3 )
    {
```



It shows the sequence of

how and where `parsub` is declared

- when you double-click on `jschar`, the next window shows how and where `jschar` is declared.

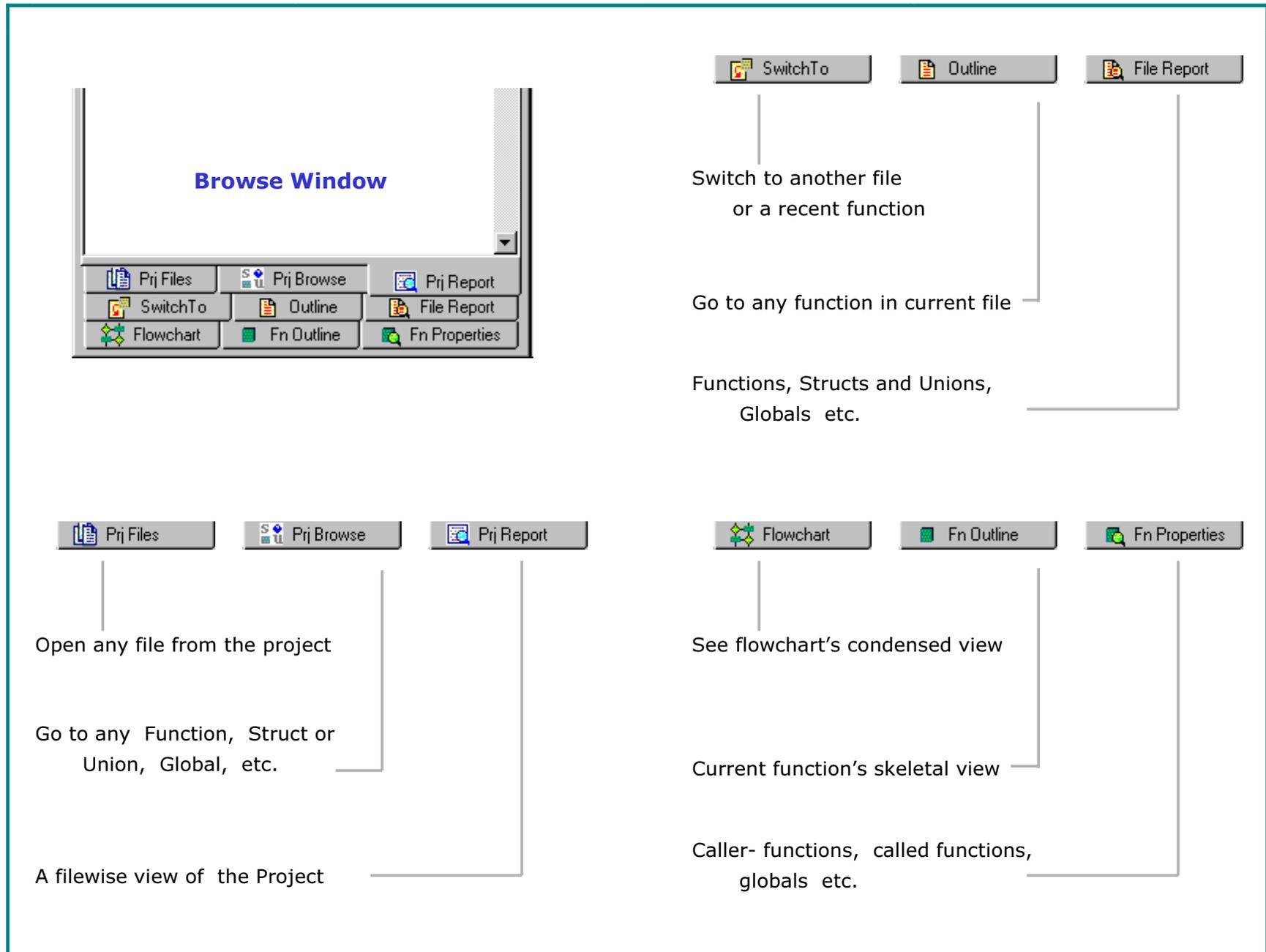


- if  is selected, a new window is opened.

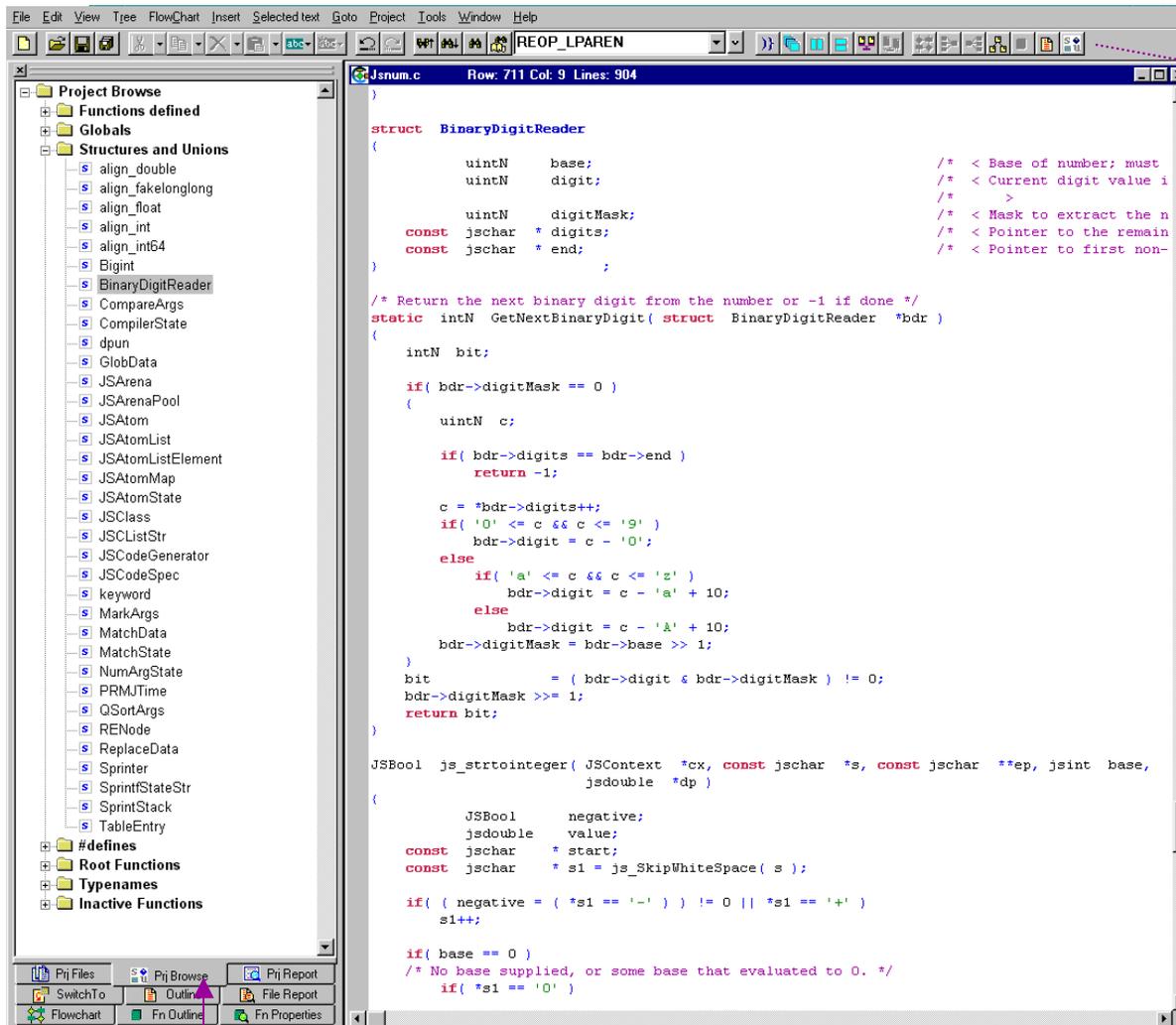
The cursor is on `parsub`

You do not have to go to the header files.

Tabs in the Browse Window



Project Browse - Go to Any Function, Struct etc.



Project-Browse

- ◆ Click the Project-Browse  icon.

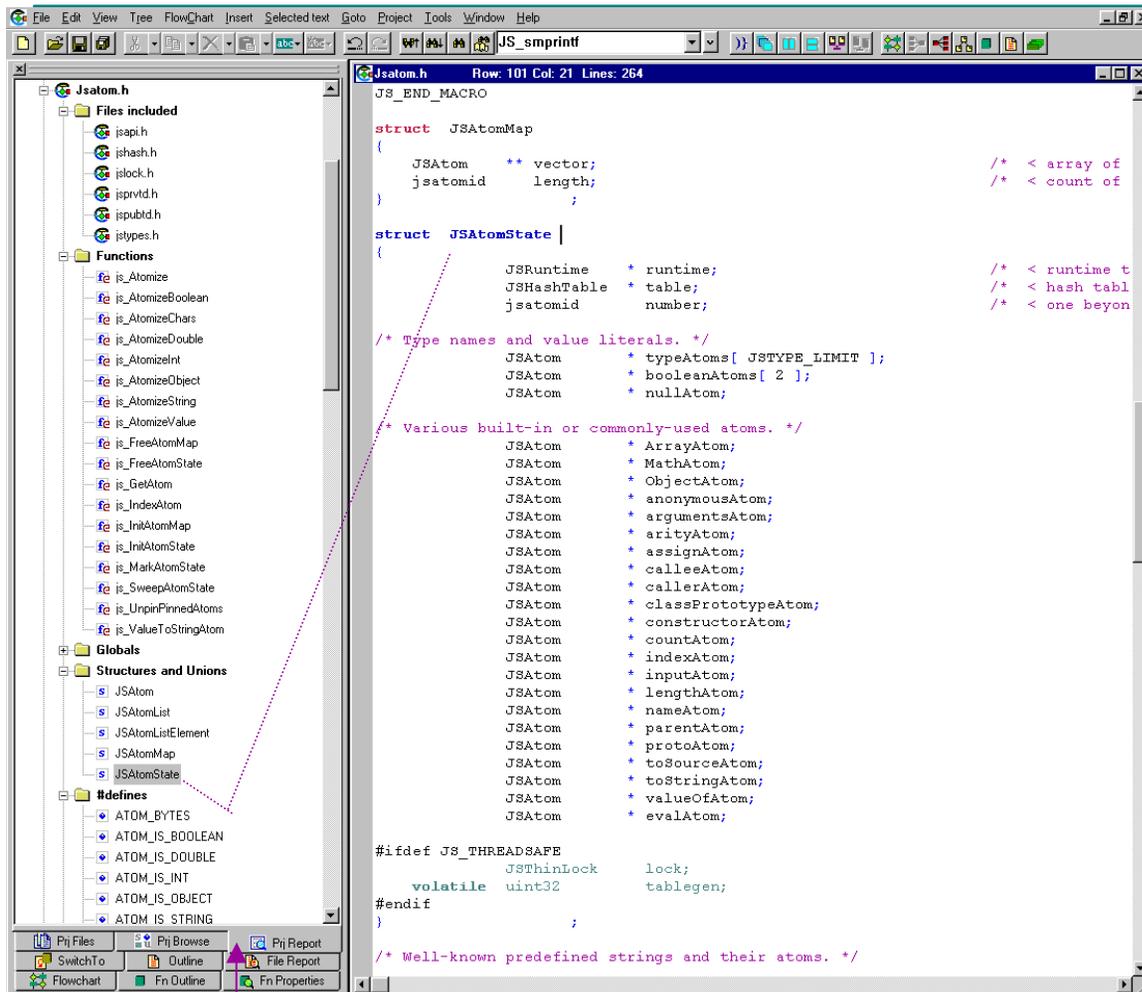
The Project-browse window appears.

- ◆ Expand any sub-tree, such as Structures and Unions

Double-click on a node to see its definition.

Also, try right-click on a node.

A Report of Globals, Functions etc. from each Source File



Project Report

- ◆ For the whole project:

click  Pj Report

to create a report for the whole project.

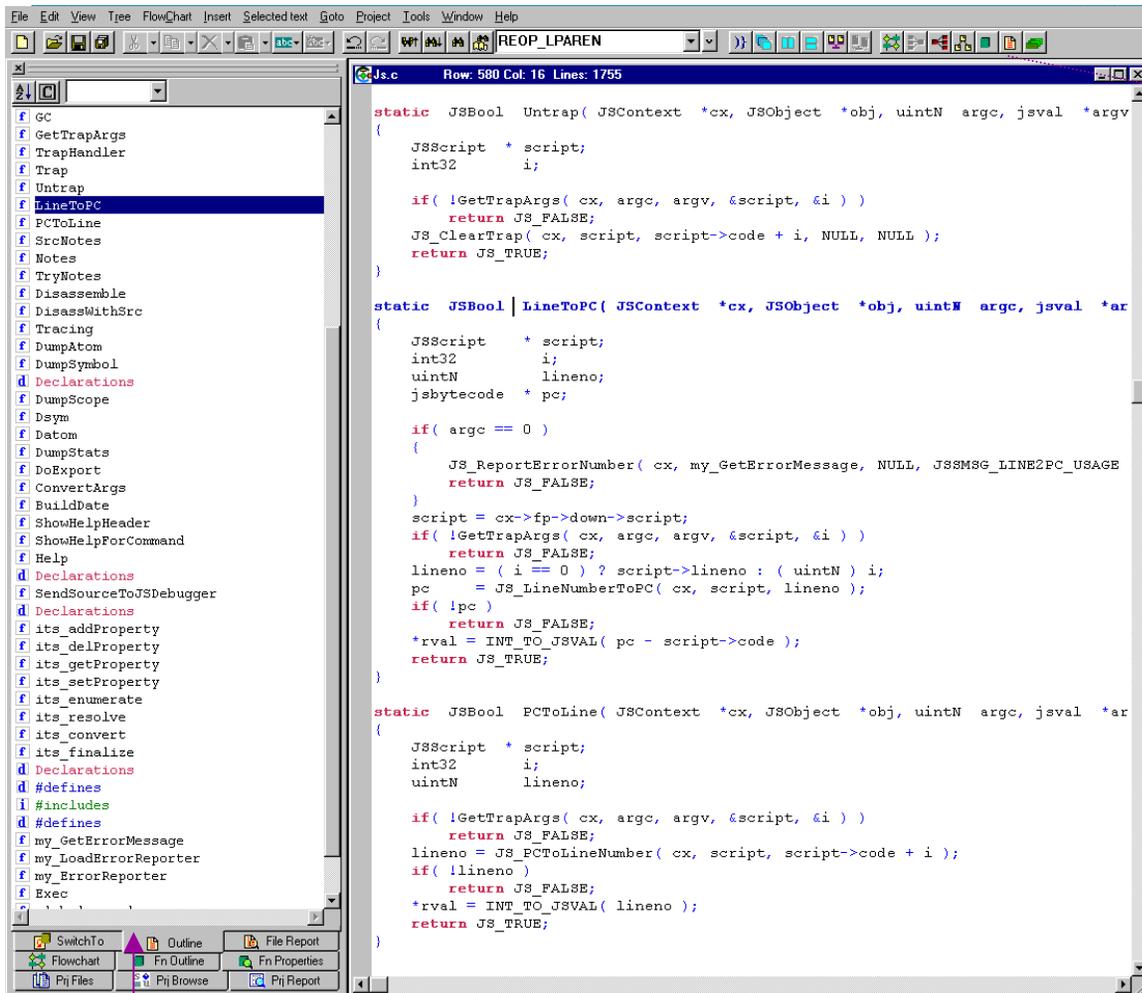
- ◆ For the current file:

click  File Report

to create a report for the current file.

← Double-click to view the occurrence in the source file.

File Outline - Go to Any Function in the Current File



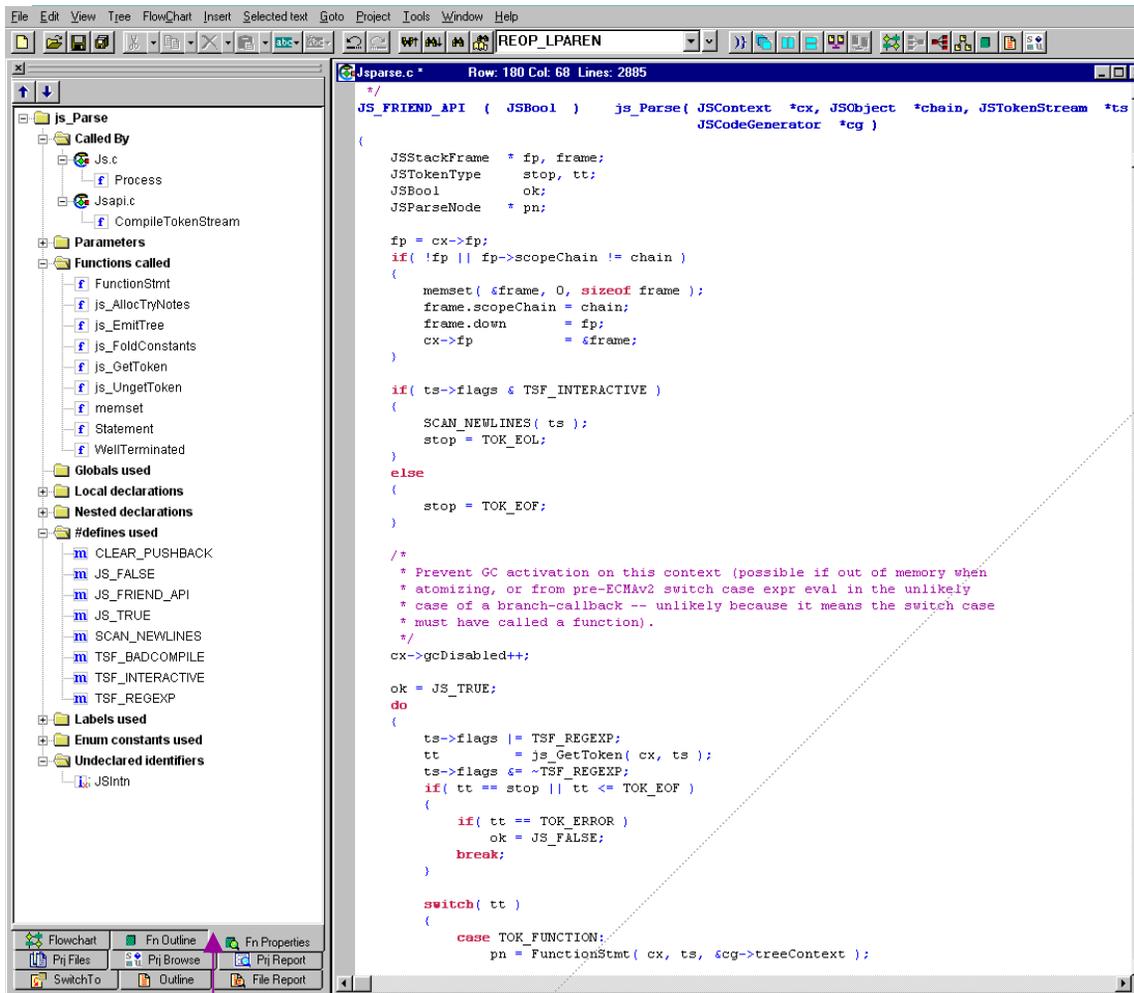
File Outline

- ◆ Click the File-outline  icon.

The File-outline window appears.

- ◆ In the File-outline window, double-click on a function-name to go to that function.

Function Properties - Called-By, Functions Called, Globals Used etc.



◆ Place the cursor in a function.

Click  **Fn Properties** tab in the browse-window.

◆ Double-click on a node.

Also, try right-click on a node.

Function Properties

Get a Clear View of the Code - Move Comments to the Right

```

c:\mozilla\js\src\jsobj.c *      Row: 1587 Col: 29 Lines: 3051
{
  JSClass      * clasp;
  JSScope      * scope;
  JSScopeProperty * sprop;

  /* Handle old bug that treated empty string as zero index.
   * Also convert string indices to numbers if applicable. */
  CHECK_FOR_INVALID_INDEX( id ) ;

  /* Lock if object locking is required by this implementation. */
  JS_LOCK_OBJ( cx, obj ); |

  /* Use the object's class getter and setter by default. */
  clasp = LOCKED_OBJ_GET_CLASS( obj );
  if( !getter )
    getter = clasp->getProperty;
  if( !setter )
    setter = clasp->setProperty;

  /* Find a sharable scope, or get a new one for obj. */
  scope = js_MutateScope( cx, obj, id, getter, setter, attrs, &sprop );
  if( !scope )
    goto bad;

  /* Add the property only if MutateScope didn't find a shared scope. */
  if( !sprop )
  {
    sprop = js_NewScopeProperty( cx, scope, id, getter, setter, attrs );
    if( !sprop )
  }
}

```

Sometimes, comments are sprinkled before and after the statements.



- ◆ Use the pull-down menu:
Edit → Move/Join Comments → Move All Comments to Comment Field

```

c:\mozilla\js\src\jsobj.c *      Row: 1587 Col: 29 Lines: 3051
{
  JSClass      * clasp;
  JSScope      * scope;
  JSScopeProperty * sprop;

  /*Handle old bug that treated empty string as zero index. */
  /* Also convert string indices to numbers if applicable. */
  /*
  CHECK_FOR_INVALID_INDEX( id ) ;

  /*Lock if object locking is required by this implementation. */
  JS_LOCK_OBJ( cx, obj ); |

  /*Use the object's class getter and setter by default. */
  clasp = LOCKED_OBJ_GET_CLASS( obj );
  if( !getter )
    getter = clasp->getProperty;
  if( !setter )
    setter = clasp->setProperty;

  /*Find a sharable scope, or get a new one for obj. */
  scope = js_MutateScope( cx, obj, id, getter, setter, attrs, &sprop );
  if( !scope )
    goto bad;

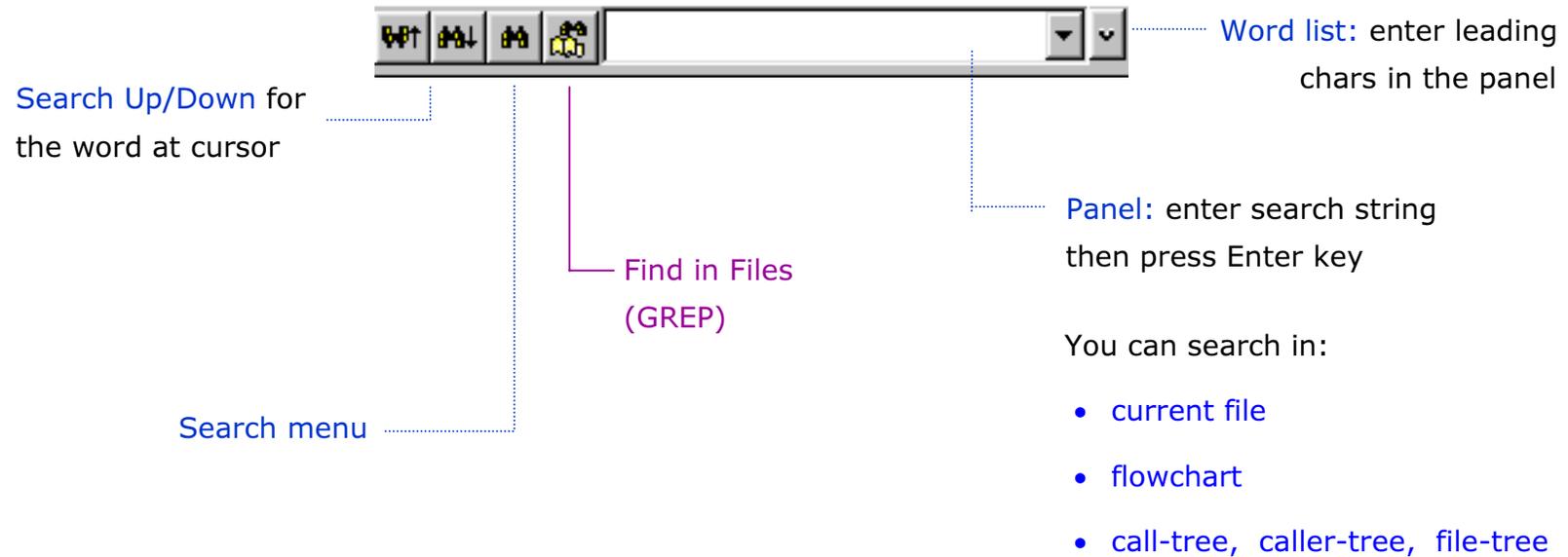
  /*Add the property only if MutateScope didn't find a
  /*shared scope.
  if( !sprop )
  {
    sprop = js_NewScopeProperty( cx, scope, id, getter, setter, attrs );
  }
}

```

You get a clear view of the code by moving comments to the right:



Find in Files (GREG) & Search in Current File



- ◆ In Search menu, use operation to show all occurrences by placing markers in the left margin of the window.

Navigating with Quick Buttons

- ◆ Scroll so that the current line goes to the top, middle, or bottom of the window.

ScrlTop	ShwDefn	Show Definition
ScrlMid	ShwDecl	Show Declaration
ScrlBott	SwtchBk	Switch back to the Previous File

- ◆ Go to the matching `)`, `]`, `}`, `*/` or `#endif`, and vice versa

MatchPair	DelPair	Delete a pair of <code>()</code> or <code>[]</code> or <code>{ }</code>
Page ↑	GoFrwd	
Page ↓	GoBack	Go back to the previous area of editing / browsing in the current file.

- ◆ Skip to the next statement that is at the same indent, e.g. from a `'case'` to the next `'case'` or skip a loop

Skip ↑	Fun ↑	Go up/down to next function
Skip ↓	Fun ↓	
Outer	Marker ↑	Go up/down to next marker (markers are placed by Search)
RptSrch	Marker ↓	

- ◆ Go to the outer indent e.g. `if`, `while`, `for`, etc.

Repeat Search

Navigating/Editing with Quick Buttons

<ul style="list-style-type: none"> ◆ Insert a line above/below the current line ◆ Move the cursor to the previous/next sentence Go to line(s) that were modified during current or recent editing 	<table border="1" style="border-collapse: collapse; width: 100%;"> <tr><td>WideWnd</td><td>FullScrn</td></tr> <tr><td>InsLn ↑</td><td>Srch ↑</td></tr> <tr><td>InsLn ↓</td><td>Srch ↓</td></tr> <tr><td>Sntnc ↑</td><td>Desel</td></tr> <tr><td>Sntnc ↓</td><td>WrdBack</td></tr> <tr><td>Mdfd ↑</td><td>DelLine</td></tr> <tr><td>Mdfd ↓</td><td>DelWord</td></tr> </table>	WideWnd	FullScrn	InsLn ↑	Srch ↑	InsLn ↓	Srch ↓	Sntnc ↑	Desel	Sntnc ↓	WrdBack	Mdfd ↑	DelLine	Mdfd ↓	DelWord	<p>Search Up/Down for the word at cursor</p> <p>Delete the preceding word (similar to Backspace)</p>
WideWnd	FullScrn															
InsLn ↑	Srch ↑															
InsLn ↓	Srch ↓															
Sntnc ↑	Desel															
Sntnc ↓	WrdBack															
Mdfd ↑	DelLine															
Mdfd ↓	DelWord															

	Flowchart	Caller-Tree	Call-Tree	File-Tree	Function Outline
MatchPair	Tile two windows vert./horiz.	Compare Files	Continue Compare		

(Intentionally Blank)