Copyright: S61G-1397, (c) COPYRIGHT

IBM C/C++ Tools

S61G-1397-00

Browser Introduction

Version 2.0

IBM C/C++ Tools

S61G-1397-00

Browser Introduction

Version 2.0

IBM

Note!

Before using this information and the product it supports, be sure to read the general information under “Notices” on page v.

First Edition (March 1993)

This edition applies to Version 2.0 of the IBM C/C++ Tools (61G1176, 61G1426) and to all subsequent releases and modifications until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

Requests for publications and for technical information about IBM products should be made to your IBM Authorized Dealer or your IBM Marketing Representative. Publications are not stocked at the address given below.

A form for readers' comments is provided at the back of this publication. If the form has been removed, address your comments to:

IBM Canada Ltd. Laboratory
Information Development
21/986/844/TOR
844 Don Mills Road
North York, Ontario, Canada. M3C 1V7

You can also send your comments by facsimile to (416) 448-6057 addressed to the attention of the RCF Coordinator. If you have access to Internet, you can send your comments electronically to **torrcf@vnet.ibm.com**; IBMLink, to **toribm(torrcf)**; IBM/PROFS, to **torolab4(torrcf)**; IBMMAIL, to **ibmmail(caibmwt9)**

If you choose to respond through Internet, please include either your entire Internet network address, or a postal address.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 1992, 1993. All rights reserved.**
Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.
IBM is a registered trademark of International Business Machines Corporation, Armonk, N.Y.

Contents

Notices v
Trademarks and Service Marks v

About This Book vii
Who Should Use This Book vii
Highlighting Conventions vii
Content and Organization vii

Introduction 1
Understanding the C/C++ Tools Browser 1
Getting Help While You Are Using the C/C++ Tools Browser . . 3
Using Contextual Help 3
Using the Online Tutorial 4
Related Publications 4

Getting Started 7
Setting Compiler Options for the Browser 7
Starting the Browser 7
Starting the Browser from the OS/2 Prompt 7
Starting the Browser from the WorkFrame/2* Environment . . 8
Browser Options 9
Cumulative Options 12
Examples on Using Browser Options 12
Browsing a Program from the Browser Control Window 12
Ending the Browser Session 13

Using the C/C++ Tools Browser Windows 15
Using the Browser Control Window 15
Using the Client Area 15
Loading and Unloading Browser Files 16
Performing a Database Search 17
Performing an Object Query 17
Selecting Window Options 18
Using the List Window 18
Using the Graph Window 20
Using the Text Window 22
The Browser Object List 24

© Copyright IBM Corp. 1992, 1993

Appendix A. Limitations and Assumptions 27
Limitations 27
Assumptions 27

Index 29

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM licensed program in this publication is not intended to state or imply that only IBM's licensed program may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Commercial Relations, IBM Corporation, Purchase, NY 10577.

Trademarks and Service Marks

The following terms used in this publication are trademarks or service marks of IBM Corporation in the United States or other countries. They are denoted by an asterisk (*) when they first appear in the text.

IBM
Operating System/2
OS/2
Presentation Manager
WorkFrame/2

© Copyright IBM Corp. 1992, 1993

About This Book

This book introduces the features of the IBM* C/C++ Tools Browser (hereafter called the C/C++ Tools browser).

Who Should Use This Book

If you are using the C/C++ Tools browser for the first time, you should read this book to become familiar with its functions. You should be familiar with C and C++ language concepts and the Operating System/2* (OS/2*) operating system.

Highlighting Conventions

Font	Convention
Bold	Names of menu-bar items, pull-down menus, pull-down choices, push buttons, and fields and selections in dialog boxes.
Monospace	Text displayed in a sample program, text or commands that you type, or text that you select.
Italics	Words that are used for emphasis and variables in command strings.

Content and Organization

Chapter 1, “Introduction,” briefly describes the C/C++ Tools browser, how to get help while using it, and lists related publications.

Chapter 2, “Getting Started,” describes the tasks you need to complete before you can start using the C/C++ Tools browser, how to start the browser, and how to end the browser session.

Chapter 3, “Using the C/C++ Tools Browser Windows,” describes the windows of the C/C++ Tools browser, and how to use them in your browsing session.

© Copyright IBM Corp. 1992, 1993

Introduction

The C/C++ Tools browser is a tool for examining programs components of application code developed in IBM 32-bit C++. It uses Presentation Manager* (PM) windowing services.

This chapter briefly describes the C/C++ Tools browser and explains how to access the online contextual help and tutorial while you are using the browser.

Understanding the C/C++ Tools Browser

The C/C++ Tools browser lets you look at your source code in many different ways. You can:

- List program components by scope (for example, class scopes), category (for example, variable), and attribute (for example, const, static, public, private)
- View relationships between program components with graphs that show class-inheritance hierarchies and map function calls
- View and edit the actual source code associated with a selected program element.

The C/C++ Tools browser is a post-compilation static analysis tool. This means that you will need to compile your source code using the **icc** command with the /Fb option before you can use the browser. The /Fb option instructs the compiler to generate the browser .BRS files for your application. You then load these files into the browser database through the Browser Control window. Loading browser files is discussed in “Starting the Browser from the OS/2 Prompt” on page 7.

© Copyright IBM Corp. 1992, 1993

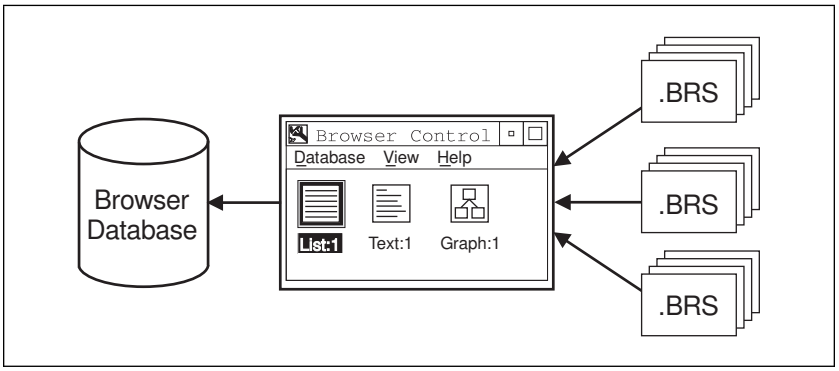


Figure 1. Loading Browser Files.. You load browser files into the browser database from the Browser Control window.

You use the C/C++ Tools browser to investigate various areas of your program by formulating *searches* and *queries* of the browser database. Searches and queries can be refined using various filters to select a specific set of program components, such as files, classes, functions, or variables, so that you can view them selectively and explore their relationships with other program components.

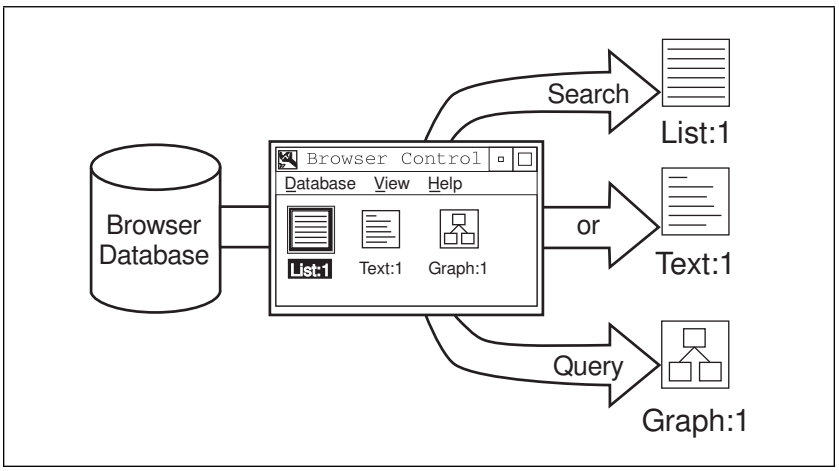


Figure 2. Browsing the Database. You can create different views of the browser database by performing searches and queries from List, Graph, and Text windows opened from the Browser Control window.

The graphical user interface consists of a control window and three types of browser windows:

The *Browser Control window* serves as the starting point for all browser use. Operations that apply to the entire browser session, such as loading and unloading browser files, are performed from the Browser Control window. You can also open existing browser windows or create new ones from this window.

A *List window* displays a list of program elements such as source files, functions, and classes. You can use filtering options in database searches and queries so that the list includes only specific program elements.

A *Graph window* displays program relationships in a graphical format, for example, all the nodes in a class hierarchy. You can specify the level of detail you want the graph to show, scroll over the graph, zoom in and out, and select program elements directly from the graph.

A *Text window* displays the source code for a selected object or program component. You can edit the actual source-code text from this window.

You can open multiple list, graph, and text windows within a single browser session from the Browser Control window.

Getting Help While You Are Using the C/C++ Tools Browser

Online help in the form of contextual help and an online tutorial gives you detailed information about using the C/C++ Tools browser.

Using Contextual Help

Documentation for the C/C++ Tools browser is available as context-sensitive help. You can ask for help on any menu choice, window, or entry field for information on how to use a particular item.

You can access browser help in one of the following ways:

Select a choice from the **Help** pull down.
Press F1 from any browser window
Press F1 while highlighting any menu bar item.
Press F1 while highlighting any pull-down choice.
Press F1 or select the **Help** push button in any secondary window

You can quickly accomplish most of the basic browser tasks using the mouse. For a list of tasks and mouse actions, refer to the online help.

You can also access the C/C++ Tools browser pull-down choices from the keyboard. The combination of keys that you use to access the pull-down choices are called *accelerator keys*. For a list of these keys, refer to the online help.

Using the Online Tutorial

If you are using the C/C++ Tools browser for the first time, you can use the online tutorial to become familiar with its functions through a guided browser session using a sample C++ payroll program called PAYROLL.CPP.

To access the tutorial, select **Tutorial** from the **Help** pull down in any of the browser windows.

Related Publications

The following publications contain information you may find useful:

IBM C/C++ Tools: C++ Language Reference, S61G-1185, presents the IBM C/C++ Tools (referred to throughout this book as C/C++ Tools) definition of the C++ programming language.

IBM WorkFrame/2: Introduction, S61G-1428, describes how to install the WorkFrame/2 product.

The following publications provide information about the C/C++ Tools product:

IBM C/C++ Tools: Programming Guide, S61G-1181, shows you how to use the C/C++ Tools product to compile, link, and run C and C++ programs on the Operating System/2* (OS/2*) operating system.

IBM C/C++ Tools: Migration Guide, S61G-1182, describes how to migrate existing C and C++ programs to the C/C++ Tools product, including migration from 16-bit to 32-bit code.

IBM C/C++ Tools: Installation, S61G-1363, describes the installation procedure.

IBM C/C++ Tools: License Information, S71G-1453, provides a summary of the C/C++ Tools features and gives warranty information.

Getting Started

This chapter describes how to prepare your program for use with the C/C++ Tools browser and how to start and end a browser session.

Setting Compiler Options for the Browser

You need to prepare your program by compiling it with the /Fb option. The /Fb option instructs the compiler to generate the necessary browser files that contain information necessary for browsing program source code. The generated browser files have the same name as the main source files of the application, with a .BRS file name extension.

Starting the Browser

You can start the C/C++ Tools browser in the same way you start any program from the OS/2 Workplace Shell by double-clicking on its icon. You can also start it from the OS/2 command prompt or from the IBM WorkFrame/2 environment.

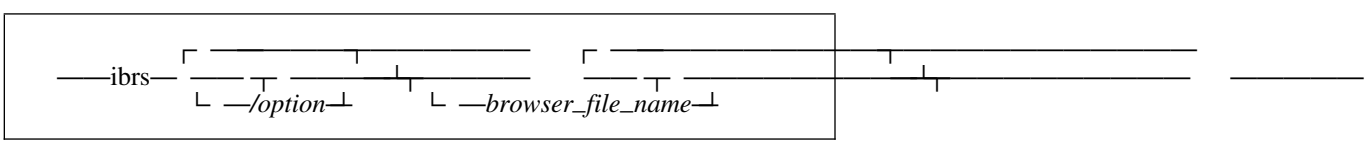
Starting the Browser from the OS/2 Prompt

From the OS/2 command prompt, enter the browser command

```
ibrs
```

then press Enter.

The syntax for the **ibrs** command is as follows:



To load one or more browser files into the browser database while starting the browser, type ibrs followed by the appropriate browser file names. You can omit the implied .BRS extension. For example,

© Copyright IBM Corp. 1992, 1993

to load three browser files for your application (mainprog.brs, subs1.brs, and subs2.brs.), type:

```
ibrs mainprog subs1 subs2
```

You can also specify the input files using OS/2 wildcards:

```
ibrs main .brs subs?.brs
```

If you type ibrs only without specifying input browser files, the browser will start without any loaded files and the Load Browser File action window is displayed automatically. To load a browser file from the Browser Control window:

1. If the Load Browser File window is not already displayed, select **Load...** from the **Database** pull-down in the Browser Control window. The Load Browser File window appears.
2. In the **Load filename** field, type the name of the browser file you want to load or select one or more browser files from the **File** list box. If your files are in a different directory, you can change the current directory by choosing a directory from the **Directory** list box.
3. Click on **Load** to close the Load window and load your selected browser files into the browser database. Browser files remain loaded until they are unloaded using the **Unload...** option.

For more information on loading and unloading browser files, refer to "Loading and Unloading Browser Files" on page 16, and the online help.

To get help on the **ibrs** command from the OS/2 prompt, type:

```
ibrs ?
```

Starting the Browser from the WorkFrame/2* Environment

To be able to compile a target program with browser information from within the WorkFrame/2 environment, you must have installed WorkFrame/2 support when you installed the C/C++ Tools files. You must have also created a project for the program you want to browse within the WorkFrame/2 environment and set the compiler options to generate browser files. You can also set C/C++ Tools browser

options within the WorkFrame/2 environment so that they remain consistent for a particular project.

For more information on creating a project, setting options, and starting tools from the WorkFrame/2 environment, see the WorkFrame/2 on-line tutorial.

Browser Options

The information from all loaded browser .BRS files is stored in the browser database for the current browser session. You can control the program components (called *facts*) of the loaded browser files that you want included in the browser database by specifying the following fact (/F) options when you start the browser:

Table 1 (Page 1 of 3). Browser Options

Option	Description	Default
/Fv["all"| "none"| "global"| "member+global"]	Facts-variable: Specifies whether *global, member+global, all,* or *none* of the facts on variables should be included in the database. For large databases, consider using *member+global*, rather than *all* to reduce the size of the database by as much as 25%.	/Fv["all"] Include *all* facts on variables in the browser database.
/Fe[+|-]	Facts-enumerator: Specifies whether *all* or *none* of the facts on enumerators should be included in the database.	/Fe[+] Include *all* facts on enumerators in the browser database.

Table 1 (Page 2 of 3). Browser Options

Option	Description	Default
/Fm[+|-]	Facts-macro: Specifies whether *all* or *none* of the facts on macros should be included in the database.	/Fm[+] Include *all* facts on macros in the browser database.
/Fd[+|-]	Facts-typedef: Specifies whether *all* or *none* of the facts on typedefs should be included in the database.	/Fd[+] Include *all* facts on typedefs in the browser database.
/X*path[;path]*	Exclude headers: Specify the names of header files whose facts you do not want to load into the database. The list of file names is kept in the database, but the facts are not. You can specify path names with the OS/2 * wildcard.	Include the facts of *all* header files in the browser database.
/B*path[;path]*	Browser file directories: Specifies where to search for the specified input .BRS files. The current directory is not searched unless specified.	Search current directory.

Table 1 (Page 3 of 3). Browser Options

Option	Description	Default
/S*path[;path]*	Source file directories: Specifies where to search for source files that are referenced in any loaded browser .BRS file if they have been moved since the browser file was generated. The directories you specify override the path as stored in the browser file.	Search for source files in the directories they were in when their respective browser files were generated.
/V[|1|2]	Verbosity: Controls messages on browser file processing. Verbosity level one (/V1) lists the name of each browser input file as it is processed. Verbosity level two (/V2) lists (1) the name of each browser input file as it is processed, (2) the number of database facts loaded for each file, and (3) the time taken to load the facts into the browser database.	/V[ ] Suppress all messages on browser file processing.

Note: You can specify browser options using either the dash form (-Fm) or the slash form (/Fm). If any but a cumulative option

is specified more than once, the last occurrence of the option is the one in effect.

Cumulative Options

The /B, /S, and /X options have a special cumulative characteristic. That is, each time you specify one of them, the option parameters you specify are appended to your previously stated parameters.

For example, the command

```
ibrs /Bc:\project1 /Ba: /Bc:\project2  .brs
```

causes the following search path to be built:

```
c:\project1;a:\;c:\project2
```

and loads all the browser input files within the search path.

Examples on Using Browser Options

Include facts on only member and global variables:

```
ibrs /Fv"member+global"  .brs
```

Exclude facts on typedefs:

```
ibrs /Fd- myprog.brs
```

Name three browser file search directories including the current directory:

```
ibrs /Bc:\project1;c:\project2;. myprog .brs
```

Specify level two verbosity:

```
ibrs /V2 myprog.brs
```

Browsing a Program from the Browser Control Window

The C/C++ Tools browser opens with the Browser Control window showing minimized List, Graph, and Text windows. You begin examining program components from the Browser Control window:

1. If you started the browser without specifying an input browser file, the Load Browser File window is displayed automatically.

Load a file from this window by following the steps outlined on page 8.

2. Once a browser file has been loaded, use an open List, Text, or Graph window by selecting its icon from the Browser Control window. You can also open a new browser window by selecting **Open** → **List**, **Open** → **Text**, or **Open** → **Graph** from the **Database** pull-down.
3. Perform an initial database search from any browser window. Browser windows and database searches are discussed in detail in the chapter on "Using the C/C++ Tools Browser Windows" on page 15, and in the online help.

Ending the Browser Session

To end the browser session, select **Close Browser...** from the **Database** pull-down in the Browser Control window, or press F3 from any browser window. The Close Browser window appears. Select one of the following options:

Click on **Yes** to end your browser session.
Click on **No** to return to the previous screen without exiting from the browser.

Using the C/C++ Tools Browser Windows

This chapter describes the different types of C/C++ Tools browser windows you use to analyze the static elements of your code. Multiple instances of browser list, graph, or text windows can exist within a single browser session.

List, graph, and text windows are opened from the Browser Control window.

Using the Browser Control Window

The Browser Control window serves as the control area of the browser; it is displayed during the entire browser session. Operations that apply to the entire browser session are invoked from this window.

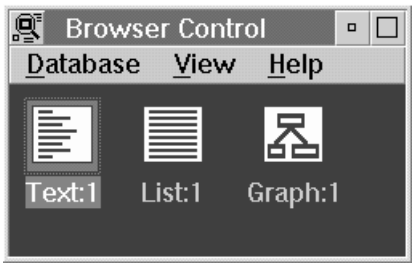


Figure 3. The Browser Control Window

Using the Client Area

Any open list, graph, or text window can be accessed from the Browser Control window. The client area of the Browser Control window displays the open browser window objects in three different layouts:

Horizontal
Arranges icons representing the window objects horizontally within the client area. This is the default layout.

© Copyright IBM Corp. 1992, 1993

Vertical
Arranges icons representing the window objects vertically within the client area.

Hierarchical
Arranges icons representing the window objects in a hierarchical format within the client area. Icons are organized by window type.

The client area of the Browser Control window contains icons that represent open browser windows in the current browser session. Icons remain visible for as long as their corresponding windows are open. To make an existing window active, double-click on its icon or name.

You can also use the browser window icons in the Browser Control window to open new List, Text, or Graph windows. Bring up the pop-up menu of an icon by positioning the mouse pointer over it and pressing mouse button 2. Select **New** from the pop-up menu to open a new window of the same type.

Loading and Unloading Browser Files

You load and unload C/C++ Tools browser files from the Browser Control window. To load or unload a browser file, select **Load...** or **Unload...** from the **Database** pull-down. You can load as many browser .BRS files as you require into the browser database at one time.

When you load a browser file, all program elements associated with it become accessible from any list, graph, or text window. The browser loads the information from each loaded browser file into its own working database. List, graph, and text windows display program components from all the loaded browser files together. Projects divided into many parts that are developed and compiled separately can be examined as a whole by loading all the browser files for the application into the browser database.

When you unload an input database file from the browser session, all open windows are reset, and the browser Object List is emptied. (The browser Object List is discussed in “The Browser Object List” on page 24.) The program components from any browser files that are still loaded remain accessible from any browser window.

Performing a Database Search

Browser windows are empty when you first open them. The first step to examining your program in a list, graph, or text window is to perform a database search. A database search displays the program components specified by your search in the window from which you invoked the search action. The Search Database action window appears automatically when you open a browser window if you have not yet performed an object query or copied an object into the Object List.

To perform a search of the database:

1. Select **Search Database...** from the **Browse** pull-down in a list, graph, or text window. The Search Database window appears.
2. Select the **Scope**, **Category**, and **Attribute** filters for the search. (A *filter* is a parameter that limits a database search or object query so that only the data that fulfills the criteria specified by the filter is retrieved. For more information on search filters, refer to the online help.)
3. Select the **Search** push button to close the window and start the search.

Performing an Object Query

Use an object query when you want get more information about a particular object in the database. The filters that you specify apply directly to the object being queried.

There are three ways to query a specific object:

Select an object from the screen then select **Query Object...** from the **Browse** pull-down. The object is copied into the Object List and the Query Object window appears with the selected element as the query object.

Double-click on an object to copy it into the Object List and bring up the Query Object window with the selected element as the query object.

Select an object and copy the object into the Object List by selecting **Copy to Object List** from the **Browse** pull-down. Then, select **Query Object...** from the **Browse** pull-down. The Query Object window appears with the last element you copied as the object of the query. This method is useful when you want to perform a query on an object that you copied from another window.

You can also scroll through the drop-down Object List in the Query Object window to select another object to query.

Selecting Window Options

You can select options to view certain attributes of the currently displayed list, graph, or text on status bars near the top of the current window using the **Status** option from the **View** pull-down of a List, Graph or Text window. Options in the **Status** cascaded menu allow you to display or hide information such as the object and filters of your last search or query, and the number of objects currently being displayed.

Clear the contents of a window by selecting **Reset** from the **View** pull-down. The window returns to its original state and behaves like a newly opened window.

Using the List Window

Use the List window to generate a list of elements from the browser database that meet the selection criteria you specify. To open a new list window, select **Open** → **List** from the **Database** pull-down in the Browser Control window then perform a database search as described on page 17.

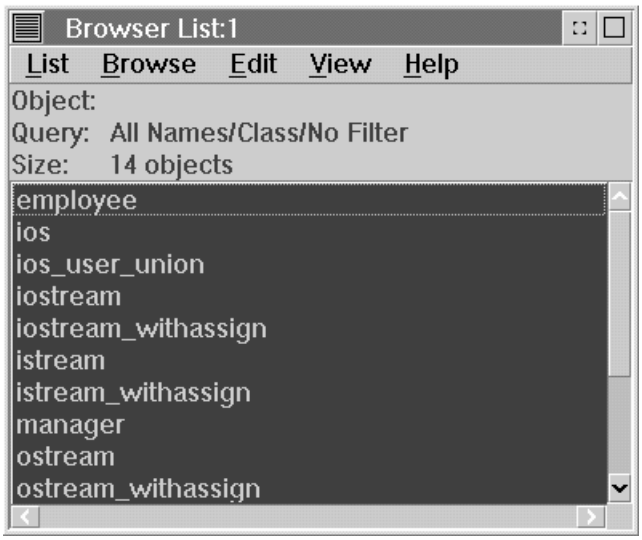


Figure 4. Browser List Window

Most of the actions you perform in the list window consist of a series of other database searches and object queries. One query usually leads to another as you continue to look at your program components in different ways. For example, you can start with a Class Scopes database search listing all the class members in the database. Then, you can focus on a specific member function and query it to find out which other functions call it.

Once a list is displayed on the window, you can query a specific list item by double clicking on the item, or by highlighting it then selecting **Query Object...** from the **Browse** pull-down. Either way, the Query Object window appears where you can specify the kind of query you want to make. You can further refine your query by specifying Scope, Category, and Attribute filters where applicable. For more information on querying an object, refer to the online help.

You can scroll over the list using the horizontal and vertical scroll bars on the bottom and right-hand side of the window. You can also use the Up and Down arrow keys and the Page Down and Page Up keys to scroll on screen at a time. You can also jump to a group of

items by typing their first letter. For example, to jump to the first of the items with a name beginning with m, type m.

You specify how much detail to display for the list items by selecting the **Qualify** and **Kind** options from the **View** pull-down. When active, **Qualify** displays the access specifier and type for each element in the list. **Kind** displays the kind or language construct (such as *variable* or *function*) of each list item.

You can save the contents of the current list to an ASCII file. From the **List** pull-down, select **Save as...** to open the Save as window where you can enter the name of the output file.

Using the Graph Window

Use the Graph window to view relationships between program components.

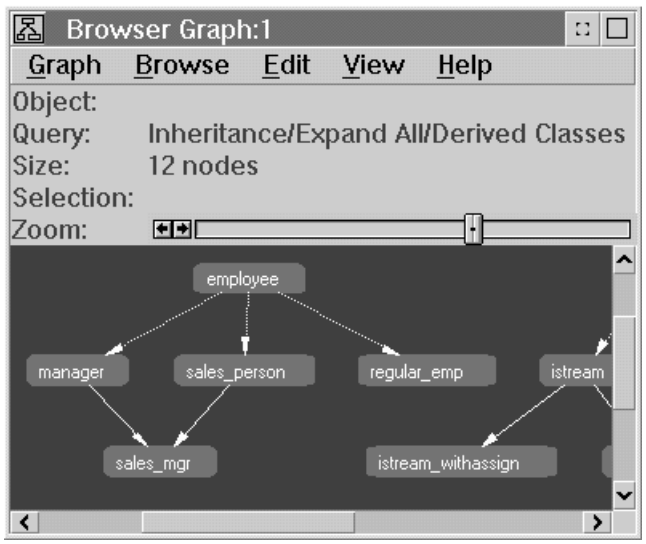


Figure 5. Browser Graph Window

The relationships are shown as nodes connected by lines. *Private*, relationships are shown using a different colored line or line style. *Virtual* base class relationships use dotted lines. Graphs with cycles, for example graphs of recursive function calls, are depicted with an arc pointing back to the source node.

As with any other type of window, the first step you perform is a database search. If there are entries in the browser Object List (the browser Object List is discussed in "The Browser Object List" on page 24), you can also start by querying an object from the list. Interactions with the Graph window consist mainly of queries on selected nodes or objects.

You can scroll around the graph using the horizontal and vertical scroll bars at the edge of the window or using the arrow keys. The **Overview...** option from the **View** pull-down is a quick and convenient way to scroll over the entire graph. When you select **Overview...**, a small window containing a representation of the entire graph is displayed at the top right-hand corner of the window. A smaller rectangle within the window represents the part of the graph currently being displayed. As you drag the viewing rectangle, the visible portion of the graph in the window changes to correspond to the new position of the rectangle. You can also change the size of the rectangle so that a smaller but magnified section of the graph is shown at one time. For more information on how to use the Overview window, refer to the online help.

You can control the detail of the graph by expanding or collapsing nodes. To expand a node means to show all its descendants; to collapse a node means to hide its descendants. To collapse a selected node, click on it once, and then select **Node** → **Collapse All** from the **View** pull-down. To expand a node, click on it once then select **Node** from the **View** pull-down. You can choose to expand the node one level (**Node** → **Expand**) or expand it entirely (**Node** → **Expand All**). For more information on expanding or collapsing a node, refer to the online help.

The Graph window offers three ways to enlarge or reduce the visible portion of the graph:

Size the viewing rectangle in the **Overview...** window.

Select **Status** → **Zoom** from the **View** pull-down to display the Zoom Slider at the top of the graph window. The Zoom Slider lets you quickly zoom in or out of the graph view while maintaining the current perspective.

Use the **Magnify** option from the **View** pull-down to magnify or reduce of a specified section of the graph. A magnification pointer allows you to select a rectangular section of the graph to expand or reduce. **Magnify** → **Area In** expands the selected rectangle to fit the graph window display. **Magnify** → **Area Out** reduces the currently displayed view of the graph to fit into the selected rectangle.

Using the Text Window

Use the Text window to display the actual source code for a list of all source files in the database, or for a focused item. First, you will need to perform a database search as described in “Performing a Database Search” on page 17.

You can view and edit the source information, however, *any changes you make will not be reflected in the browser database until you recompile your program and reload the updated browser files*.

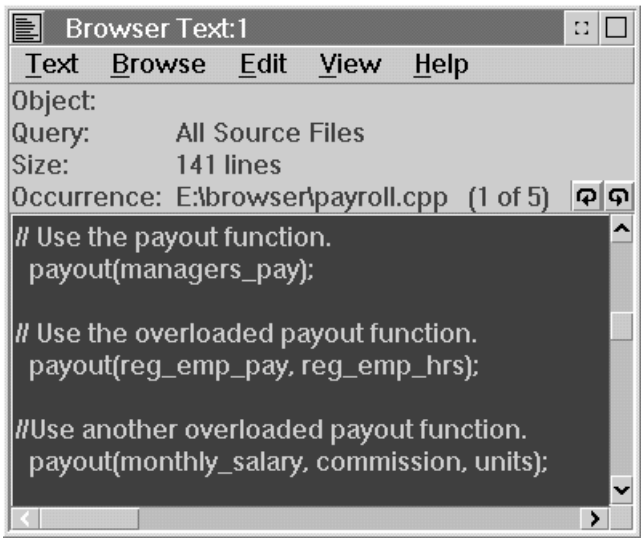


Figure 6. Browser Text Window

The result of a database search or object query from a text window is a list of what are called *occurrences*. An *occurrence* can be a source-code file, or an instance of a browser object in a source file depending on the action you perform. The first occurrence in the list is displayed in the Text window.

Table 2. What is an Occurrence?

When the action is a...	***The occurrences list is...***
Database Search	A list all of source-code files in the database.
Object Query	A list of instances where the query object appears in one or more source files in the database.

Many source-code files can be viewed from a single Text window. You select an occurrence from the list to view in the Text window using the **Occurrences** → **List** option from **File** pull-down. You can also rotate through the sequence of occurrences using the rotation controls, **Occurrences** → **Next** and **Occurrences** → **Previous**.

A useful text window feature is the **What is** option in the **View** pull-down. If you highlight a word in the text and select **What is**, the definition of the selected item is displayed in a pop-up window. If the selected string is not a valid object in the program database, a warning message appears. If the selection is ambiguous, the **What is** window displays a list of possible uses for the element.

To edit source files in the Text window, you need to deactivate the **Browse only** option in the **Edit** pull-down. Standard editing functions such as Cut and Paste, Find, and Replace are provided in the Text window. (For a complete list of editing tasks and keys, refer to the online help.)

Note: If you edit your source, you must refresh the browser database for your changes to be reflected in future searches and queries from List, Graph, and Text windows. In particular, object queries are no longer possible from the Text window if you add or delete lines in the source because facts in the database depend on their exact line placement in the source-code text at the time the browser file was generated.

To refresh the browser database after editing the source:

1. Save any changes you have made to the source files.
2. Recompile the source files from another OS/2 command prompt window.
3. From the Browser Control window, select **Load...** from the **Database** pull-down and re-load all the previously loaded browser files.

The Browser Object List

To query an object between browser windows, you must first copy it into the Object List. The browser Object List contains a list of objects you copy or query from any browser window. You can use the Object List to query an object that is not currently displayed on the active window but has been copied into the Object List from another window.

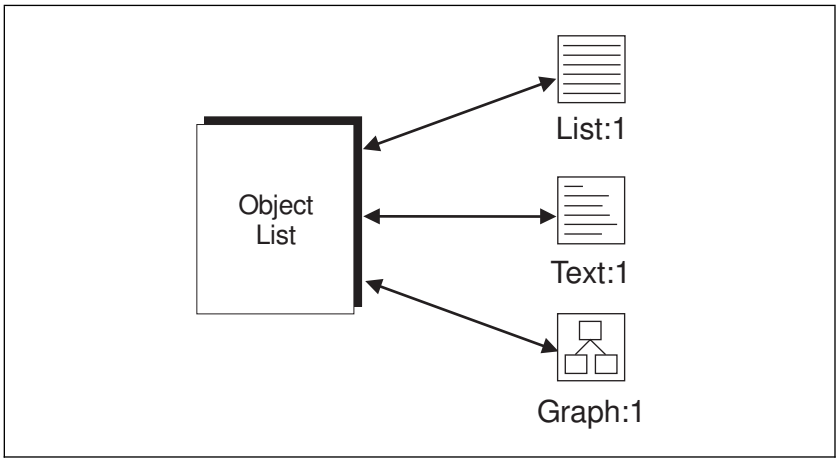


Figure 7. The Object List is shared among all browser windows.

There are three ways to copy a displayed object into the Object List:

Select a displayed object then select **Copy to Object List** from the **Browse** pull-down in a list, graph, or text window.
Select a displayed object then select **Query Object...** from the **Browse** pull-down in a list, graph, or text window.
Double-click on the object to copy it into the Object List and bring up the Query Object window with the selected item as the query object.

You can query any object in the Object List from another window of any type, or from the same window. To query an object in the Object List, select it from the Object entry field in the Query Object window (click on the arrow at the rightmost side of the field to display the Object List).

The browser Object List remains intact until the C/C++ Tools browser is closed or until a browser file is unloaded from the browser session. In both cases, the Object List is cleared of its contents.

Appendix A. Limitations and Assumptions

This appendix describes the limitations and assumptions of the C/C++ Tools browser.

Limitations

The C/C++ Tools browser has the following restrictions:

- The browser Object List holds only the last 25 objects copied or queried.
- The browser allows a total of 19 open browser windows per browser session.
- The browser List window container has a limit of 64KB. If this limit is exceeded, the first 64KB worth of data is retained and a warning message is displayed.
- The /B, /X, and /S options have a limit of 100 path entries. If more than 100 are specified, the first 100 path entries are used and a warning message is displayed.
- You can name up to 25 browser files to load on the command line. There is no restriction on the number of browser files you can load from the Browser Control Window.

Assumptions

The C/C++ Tools browser assumes that:

- Source files have not been renamed since their corresponding browser files were generated.
- If source files have been moved since their corresponding browser files were generated, their new directory or directories are specified using the /S option.
- The PATH, HELP, DPATH, and LIBPATH environment variables contain the following paths, assuming you installed the C/C++ Tools product on drive C: and used the IBM-supplied defaults for the installation:

© Copyright IBM Corp. 1992, 1993

```
PATH=C:\IBMC\BIN
HELP=C:\IBMC\HELP
DPATH=C:\IBMC\DLL;C:\IBMC\SYS
LIBPATH=C:\IBMC\DLL
```

These environment variables are set with the appropriate paths during installation.

Index

A

ASCII file 20
attribute 19
attribute filter 17

B

books you should read 4
browser
  ending 13
  examples on starting 12
  introduction 1
  options for starting 9
  starting from the OS/2 prompt 7
  starting from the WorkFrame/2 environment 8
browser database
  components to include in 9
  loading .BRS files into 7
  refreshing 22
browser files
  generating 7
  loading 7, 8, 16
  unloading 16

C

category 19
category filter 17
compiler options 7
compiling a program 7
control window
  browsing from 12
  client area 15
  description 3, 15
  icon layout 15
  loading .BRS files from 16
control window *(continued)*
  opening windows from 18
  unloading .BRS files from 16

D

database
database search 17, 19, 21, 23

F

facts, include in database 9
filters
  attribute 17, 19
  category 17, 19
  scope 17, 19

G

graph window
  collapsing nodes 21
  description 3, 20
  expanding nodes 21
  magnify 22
  overview 21
  ways to zoom 21
  zoom slider 22

H

help on the browser
  command line 8
  contextual 3
  online tutorial 4

© Copyright IBM Corp. 1992, 1993

I

ibrs command
- examples 12
- loading browser files with 7
- syntax 7

icon layout
- hierarchical 16
- horizontal 15
- vertical 16

K

kind 20

L

list window
- description 3
- overview 18
- saving contents of 20
- scrolling 19

loading browser files 7, 8, 16

M

magnify 22

N

node
- collapsing 21
- expanding 21

O

object list
- copying an object into 18, 25
- description 24
- emptied at unload 17

object query 17, 19, 23, 25
occurrences 23
options
- compiler 7

options for starting the browser
- cumulative 12
- table of 9

options, command line
- examples 12

overview window 21

P

preparation tasks 7

Q

qualify 20
query 17, 19, 23, 25

R

reset option 18

S

save as 20
scope 19
scope filter 17
scroll bars 19, 21
scrolling 21
search 17, 19, 21, 23
selecting options 18
shortcut keys 4
status option 18

T

text window
- description 3, 22
- occurrences 23
- what is 24

U

unloading browser files 16, 25

W

what is 24
windows
  browser control 3, 15
  graph 3, 20
  list 3, 18
  resetting 18
  status bar 18
  text 3, 22
WorkFrame/2
  starting the browser from 8

Readers' Comments — We'd Like to Hear from You

IBM C/C++ Tools
Browser Introduction

Version 2.0

Publication No. S61G-1397-00

Please use this form only to identify publication errors or to request changes in this publication. Direct any requests for additional publications, technical questions about IBM systems, changes in programming support, and so on, to your IBM representative or to your nearest IBM branch office. You may use this form to communicate your comments about this publication, its organization, or subject matter **with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.**

If your comment does not need a reply (for example, pointing out a typing error), check this box and do not include your name and address below. If your comment is applicable, we will include it in the next revision of the manual.

If you would like a reply, check this box. Be sure to print your name and address below.

You can also send your comments by facsimile to (416) 448-4414 addressed to the attention of the RCF Coordinator. If you have access to Internet, you can send your comments electronically to **torrcf@vnet.ibm.com**; IBMLINK, to **toribm(torrcf)**; IBM/PROFS, to **torolab4(torrcf)**; IBMMAIL, to **ibmmail(caibmwt9)**.

If you choose to respond through Internet, please include either your entire Internet network address, or a postal address.

Page number(s): Comment(s):

Name

Address

Company or Organization

Phone No.

PLACE
POSTAGE
STAMP
HERE

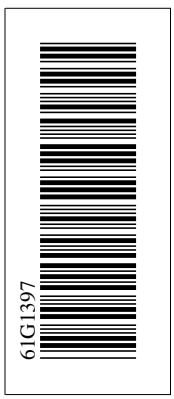
IBM Canada Ltd. Laboratory
Information Development
21/986/844/TOR
844 DON MILLS ROAD
NORTH YORK ONTARIO CANADA M3C 1V7

Fold and Tape **Please do not staple** Fold and Tape

Readers' Comments — We'd Like to Hear from You
S61G-1397-00

IBM®

IBM®

Part Number: 61G1397
Program Number: 61G1176
61G1426

Printed in U.S.A.