Contents

Introduction

TracePlus displays the interaction between a Windows 3.1 or Win32 application and the API in real time. TracePlus displays API constants, structures, buffers, and flags formatted as in the API specification. This help is for TracePlus/ODBC, TracePlus/SQL Server, TracePlus/VBX+Windows, and TracePlus/Winsock (both 16 and 32 bit versions).

Notes for 32 bit versions of TracePlus

- 32 bit versions of TracePlus use the standard Microsoft defined debugging interface for Win32 applications. The Microsoft interface does not permit detaching from a process after debugging has started. Therefore, if TracePlus is terminated, it will also terminate all processes being traced. Stopping the trace will not detach TracePlus from a process.

Be sure that if you are tracing a server application, leave the TracePlus32 application running after stopping the trace, or terminate the server application, and restart it (be sure that TracePlus32 is not running or the trace is stopped before starting the server application again).

- TracePlus32 can trace a Win32 console application. To do this, launch the console application from TracePlus32. See the topic **Setting Filters** for more details on this procedure.
- TracePlus32 does not trace Win32 applications started before TracePlus32, unless they are explicitly selected from the *Tasks* listbox. See the topic **Setting Filters** for more details on this procedure.
- TracePlus32 does not trace 16 bit Windows applications.

Getting Started

Running Traces

General Procedures

Note that these procedures are listed in alphabetical order.

TracePlus/ODBC-Specific Procedures

TracePlus/Winsock-Specific Procedures

TracePlus/SQL Server-Specific Procedures

TracePlus/VBX-Specific Procedures

References

Running TracePlus for the First Time

The first time you run TracePlus/Winsock, follow the procedures below:

1 In the Help menu, select View README.WRI.... This command provides information that has become available since the manual was printed,
and other vital information.

2 Select Options... (Trace menu) to set your trace preferences. Alternatively, press **Ctrl** + **O**, or select the Trace icon in the tool bar.

3 Select Filters... (Trace menu) to set trace functions and tasks**.** Alternatively, press **Ctrl** + **F**, or select the Filters icon in the tool bar.

TracePlus may open with the title bar and menu bar hidden.   To display the title bar and menu bar, press **Alt** + **/**.

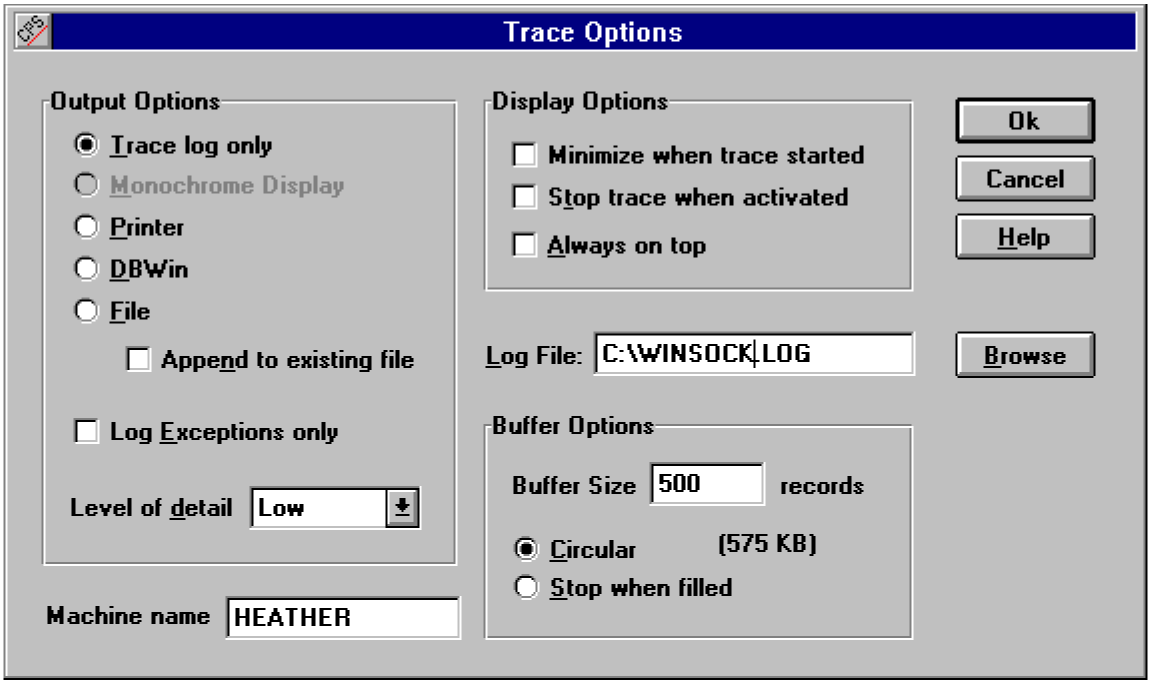
To access help from the TracePlus window, press **F1**.

Setting Your Trace Preferences

To set your trace preferences, select Options... in the Trace menu, or press **Ctrl** + **O**.

Shortcut: Select the Trace icon.

The Trace Options dialog box is displayed. Select any field or button for more information on using this dialog box.



Output Options

● Trace log only
○ Monochrome Display
○ Printer
○ DBWin
○ File
□ Append to existing file

□ Log Exceptions only

Level of detail: Low

Machine name: HEATHER

Display Options

□ Minimize when trace started
□ Stop trace when activated
□ Always on top

Log File: C:\WINSOCK.LOG

Buffer Options

Buffer Size 500 records

● Circular (575 KB)
○ Stop when filled

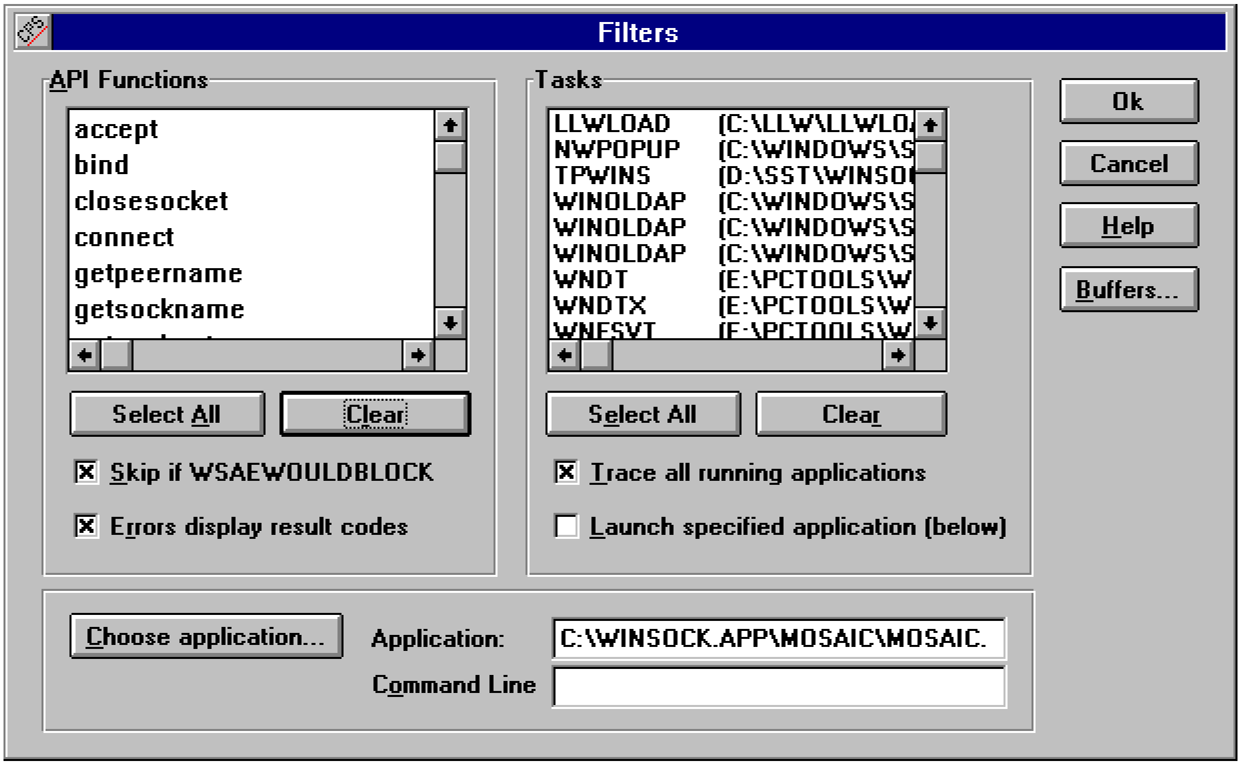
Select the Ok button, or press **Enter** to set your preferences and exit the dialog box.

Setting Filters

To set your trace preferences, select Filters... in the Trace menu, or press **Ctrl** + **F**.

Shortcut: Select the Filters icon.

The Filters dialog box is displayed. Select any field or button for more information on using this dialog box. Note that the dialog box may be slightly different, depending on your TracePlus application.



API Functions:

```
accept
bind
closesocket
connect
getpeername
getsockname
```

Tasks:

```
LLWLOAD     [C:\LLW\LLWLO
NWPOPUP     [C:\WINDOWS\S
TPWINS      [D:\SST\WINSO
WINOLDAP    [C:\WINDOWS\S
WINOLDAP    [C:\WINDOWS\S
WINOLDAP    [C:\WINDOWS\S
WNDT        [E:\PCTOOLS\W
WNDTX       [E:\PCTOOLS\W
WNFSVT      [E:\PCTOOLS\W
```

☒ Skip if WSAEWOULDBLOCK

☒ Errors display result codes

☒ Trace all running applications

☐ Launch specified application (below)

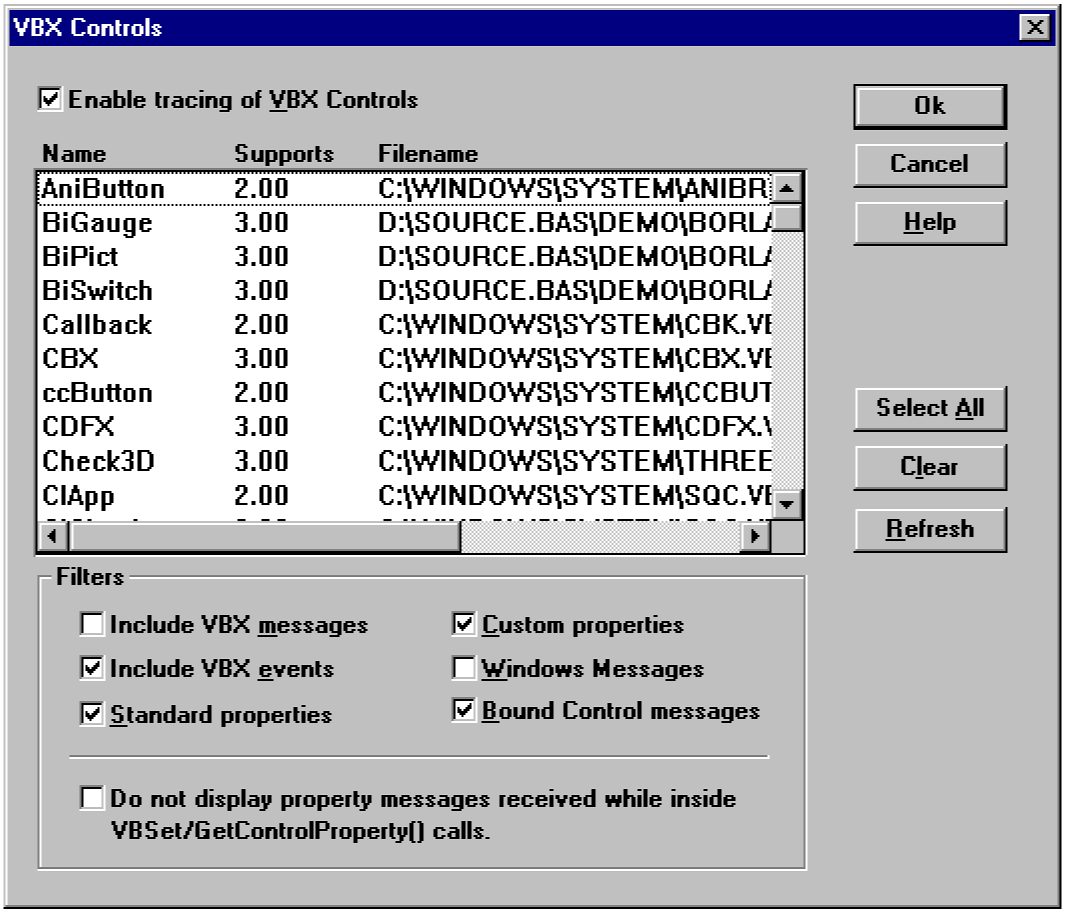
Application: C:\WINSOCK.APP\MOSAIC\MOSAIC.

Command Line:

VBX Controls

You can selectively enable or disable specific VBX controls from being traced. This feature prevents the trace log from filling up with trace records from VBX controls you are not interested in.

The following dialog box is displayed. Click on option fields and buttons for more information about how to use this dialog box.



☑ Enable tracing of VBX Controls

Name	Supports	Filename
AniButton	2.00	C:\WINDOWS\SYSTEM\ANIBR
BiGauge	3.00	D:\SOURCE.BAS\DEMO\BORL
BiPict	3.00	D:\SOURCE.BAS\DEMO\BORL
BiSwitch	3.00	D:\SOURCE.BAS\DEMO\BORL
Callback	2.00	C:\WINDOWS\SYSTEM\CBK.V
CBX	3.00	C:\WINDOWS\SYSTEM\CBX.V
ccButton	2.00	C:\WINDOWS\SYSTEM\CCBUT
CDFX	3.00	C:\WINDOWS\SYSTEM\CDFX.
Check3D	3.00	C:\WINDOWS\SYSTEM\THREE
ClApp	2.00	C:\WINDOWS\SYSTEM\SQC.V

Filters

□ Include VBX messages
☑ Include VBX events
☑ Standard properties
☑ Custom properties
□ Windows Messages
☑ Bound Control messages

□ Do not display property messages received while inside VBSet/GetControlProperty() calls.

Specifying the API Help File Location

Select API Help File... (File menu) to specify the location of the API help file, if this file exists on your system. Specify the file as in a standard Open dialog box.

Once you specify the location of the API help file, you can access API help while the debugger is running. For more information on accessing API help, see Getting API Help.

TracePlus/VBX+Windows has the option of specifying two different help files:

- The Windows 3.1 SDK help file
- The Visual Basic Control Development Kit help file

Prepending File Name and Line Number

You can embed **SSTPrintf()** and **SSTPrintfEx()** in you program. You can then specify that these formatted text strings be included in the log file. **SSTPrintfEx()** formats the string with the filename and line number prepended to the string.

Starting a Trace

To start a trace, select Start in the Trace menu, or press **Ctrl** + **S**.

▶

Shortcut: Select the Start icon.

Before you start the trace, set you trace preferences and filters.

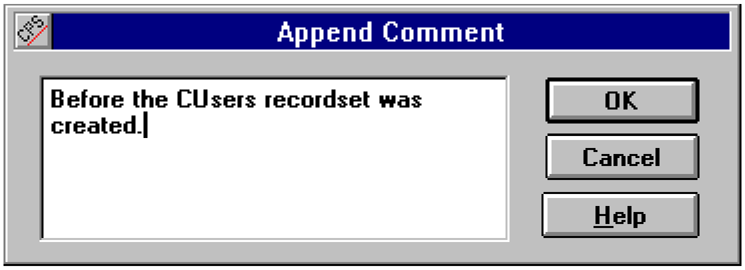
Note that you can drag single files from the Windows File Manager and drop them on the TracePlus window or the TracePlus icon. The specified *.EXE* file will be launched when the trace is started.

Appending a Comment

Select Append Comment... (Trace menu) to add a comment to the end of a trace while the trace is running.

Shortcut: Select the Comment icon, or press **Ctrl** + **Shift** + **A**.

A dialog box similar to the following is displayed.



Before the CUsers recordset was created.

In the text box, type your comment and select the OK button. To exit the dialog box without appending the comment, select the Cancel button.

Pausing a Trace

To pause a trace, select Pause in the Trace Menu, or press **Ctrl** + **A**.

Alternatively, If you selected the **Stop trace when activated** option in the Trace Options dialog box, tracing will stop when you activate the TracePlus program.

●

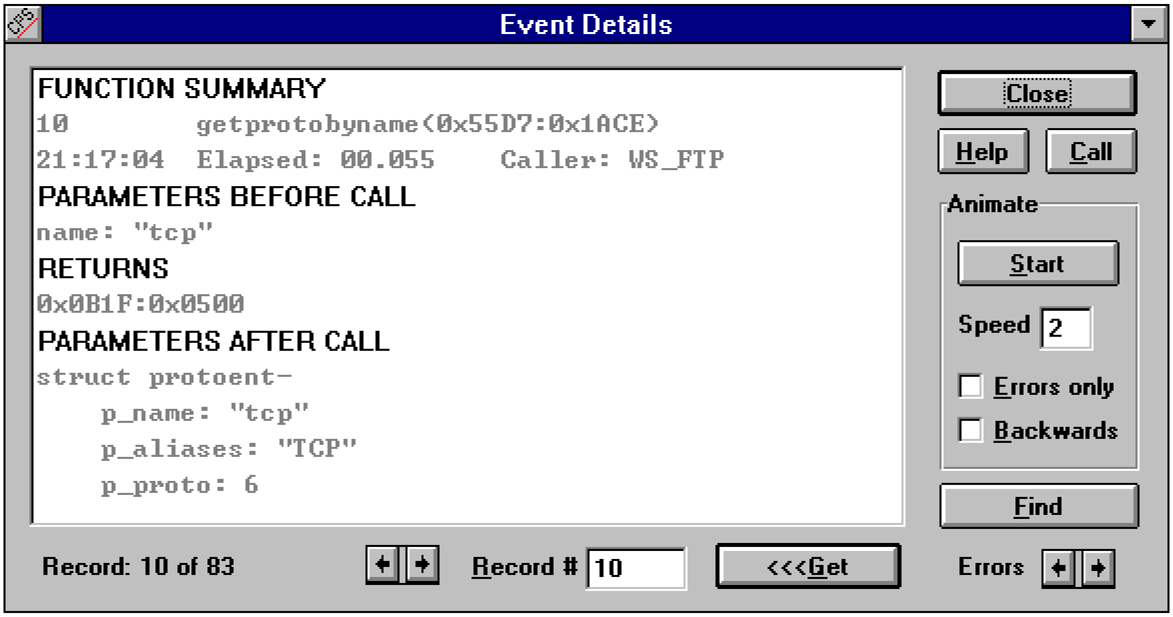
Shortcut: Select the Stop icon.

Viewing Trace Event Details

Select Event Details (Trace menu) to view details of trace events. Alternatively, double click on a trace record in the TracePlus window, or press **Ctrl** + **E** .

Shortcut: Select the Details icon in the tool bar.

The Event Details dialog box similar to the one below is displayed. Click on the fields and buttons below for more information on how to use this dialog box.



```
FUNCTION SUMMARY
10        getprotobyname(0x55D7:0x1ACE)
21:17:04  Elapsed: 00.055    Caller: WS_FTP
PARAMETERS BEFORE CALL
name: "tcp"
RETURNS
0x0B1F:0x0500
PARAMETERS AFTER CALL
struct protoent-
    p_name: "tcp"
    p_aliases: "TCP"
    p_proto: 6
```

Speed 2

Record: 10 of 83

Record # 10

The Event Details dialog box can remain on the screen as long as the trace buffer is not cleared either by selecting the Reset Events item in the Trace menu or selecting the Clear icon.

Continuing a Trace

Select Continue in the Trace menu after you have stopped/paused the trace to save the current trace and continue tracing events.

▷

Shortcut: Select the Continue icon.

Clearing the Trace Log

To clear the trace log and buffers, select Reset Events in the Trace menu.

🗑

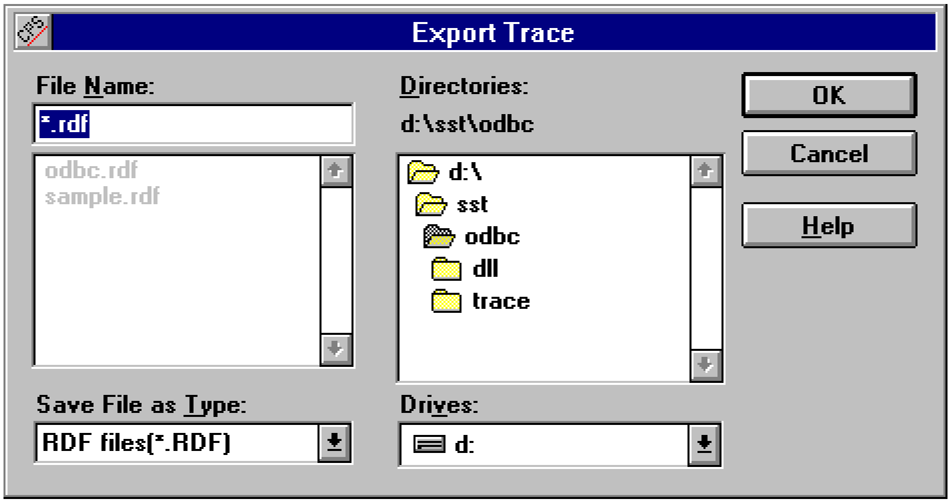
Shortcut: Select the Clear icon.

The Event Details dialog box is displayed, and then closes.

Exporting Trace Log Files

Select Export Trace... in (Trace menu) to export trace log files from a customer site to a file that can be imported into TracePlus running on your PC. This allows you to debug a customer's system.

The Export Trace dialog box is displayed.



Export Trace

File Name: *.rdf

Directories: d:\sst\odbc

Save File as Type: RDF files(*.RDF)

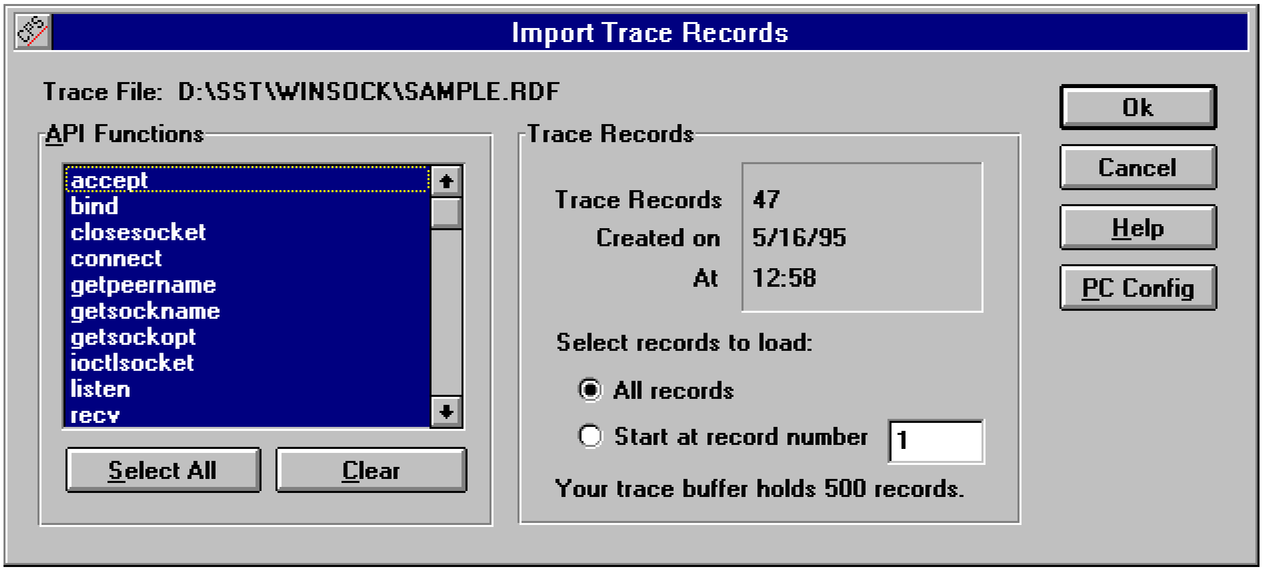
Drives: d:

Specify the destination for the *.RDF* file as in a standard Save As dialog box.

Importing Trace Log Files

Select Import Trace... (Trace menu) to import trace logs from another copy of TracePlus, such as one from a customer site that has been exported as a *.RDF* file.   This feature allows you to load traces made on other PCs into TracePlus. Once the log is loaded, you can view and manipulate it as if the trace was generated on your own PC..

When you load an exported *.RDF* file onto your system and select Import Trace..., a dialog box similar to the following is displayed. This dialog box displays API functions and records from the remote PC. Select buttons, options, and fields for more information on using this dialog box.


Import Trace Records

Trace File: D:\SST\WINSOCK\SAMPLE.RDF

Trace Records

Trace Records 47

Created on 5/16/95

At 12:58

Select records to load:

● All records

○ Start at record number 1

Your trace buffer holds 500 records.

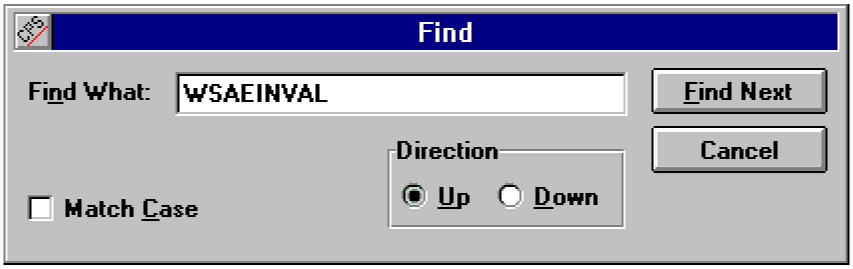
Exiting TracePlus

Select Exit in the File menu or press **Alt** + **F4**.

Finding a Trace Record

To find a record in the Event Details dialog box, select the Find button.   To search for strings in the trace log from the main screen, Select Find... (Trace menu) or press **F3**.

The Find dialog box is displayed.



Find What: WSAEINVAL

Direction ● Up ○ Down

□ Match Case

You can search by any numeric or alphabetic characters.   If you search alphabetically, you can do a case-sensitive search (if the word for which you are searching contains capital letters) by selecting the Match Case option.   You can also search up or down through records.

Getting API Help

Before you access API you must first specify the location of the API help file by selecting API Help File... (File menu) and specifying the file in the API Help File dialog box.

Select *[API_Name]* API... (Help menu) to access help on the API.

Shortcut: Select the API Help icon, or press **Ctrl** + **F1**.

Select Specific *API_Name* Function (Help menu) to access help on a specific API function.

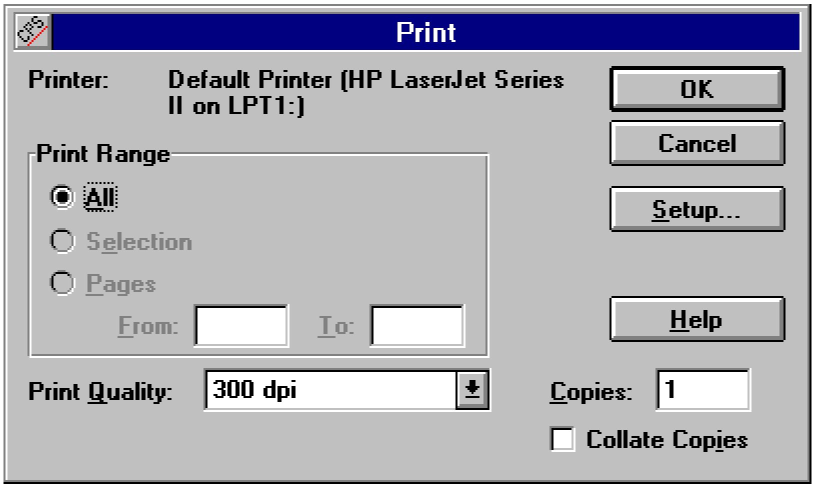
Shortcut: Select the API Function Help icon, or press **Shift** + **F1**.

Note that the public domain *WINSOCK.HLP* is included with TracePlus/Winsock and has hypertext links to the API functions that appear in the trace log. By pressing **Shift** + **F1** in the main window, or pressing the Call button in the Details dialog box, you can get detailed help on that Winsock function.Select Specific *API_Name* Function (Help menu) to access help on a specific API.

Printing the TracePlus Log

Select Print in the File menu, or press **Ctrl** + **P**, or to print the contents of the screen.

Shortcut: Select the Print icon.



Printer: Default Printer (HP LaserJet Series II on LPT1:)

Print Range

● All

○ Selection

○ Pages

From: To:

Print Quality: 300 dpi

Copies: 1

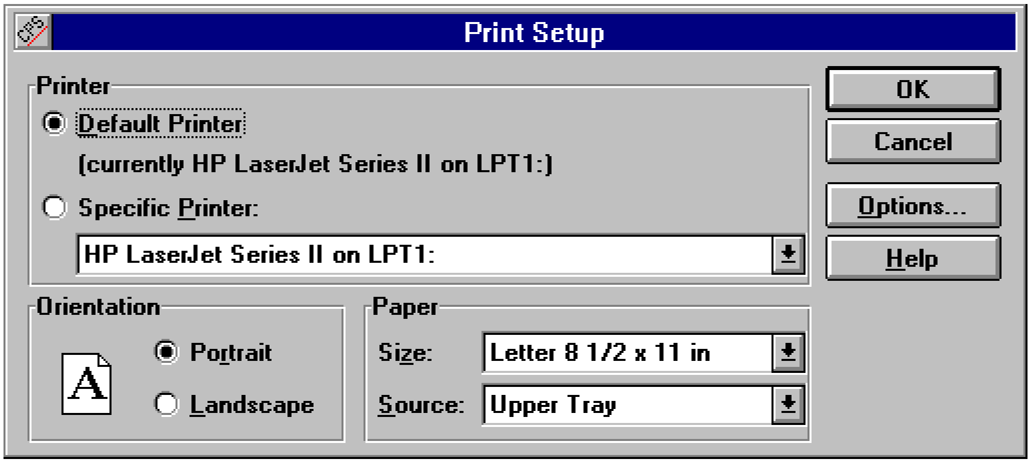
□ Collate Copies

The default print range is all. You cannot choose the Selection or Pages options. You can select the print quality, the number of copies you want to print, and if you want the copies collated.

Select the Setup... button to access the printer options available to you from the printer manufacturer.

Setting up Your Printer

Select Printer Setup in the File Menu to access the Print Setup dialog box and set up your printer.



Print Setup

Printer

● Default Printer

(currently HP LaserJet Series II on LPT1:)

○ Specific Printer:

HP LaserJet Series II on LPT1:

Orientation

● Portrait

○ Landscape

Paper

Size: Letter 8 1/2 x 11 in

Source: Upper Tray

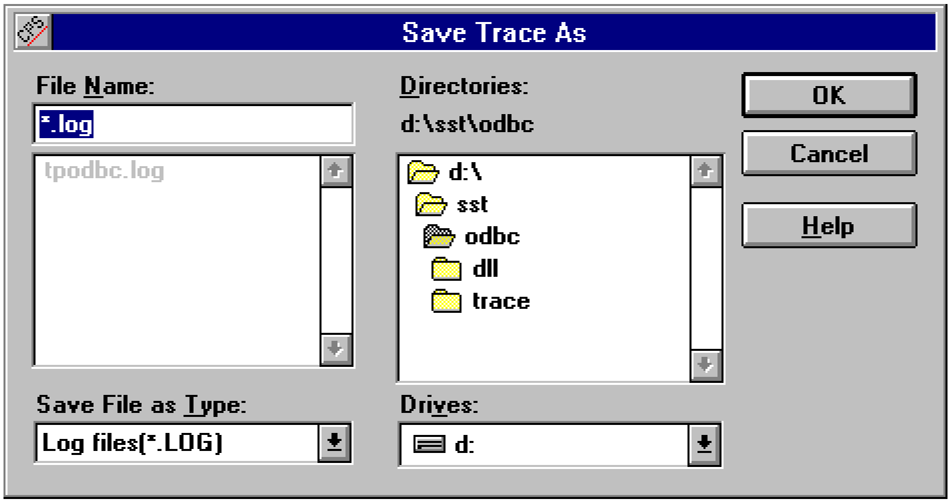
Select the Options... button in the dialog box to access the printer options available to you from the printer manufacturer.

Saving the Configuration

Select Save Configuration (Configure menu) to save the window position, Trace Options dialog box settings, and the Filters dialog box settings (except for tasks, or applications).

Saving Trace Information

If you did not specify the log file in the Trace Options dialog box, you can save the contents of a trace buffer in ASCII format to a file by selecting Save Trace As... (File menu). The following dialog box is displayed.



File Name: *.log

Directories: d:\sst\odbc

Save File as Type: Log files(*.LOG)

Drives: d:

Specify the log file as in a standard Save As dialog box.

Using SSTPrintf and SSTPrintfEx

TracePlus provides two functions, **SSTPrintf()** and **SSTPrintfEx()**,that allow you to insert your own messages into the trace log. Both functions have the same functionality as the C function **printf()**. Additionally, if you are running two or more TracePlus applications, you can specify which TracePlus application will log your message.

Description

You need the following files to use **SSTPrintf()** and **SSTPrintfEx()**:

- *SSTDEBUG.H*: This header file contains the prototypes for the functions.
- *SSTDEBUG.LIB*: This library file includes the exports for *SSTDEBUG.DLL*.
- *SSTDEBUG.DLL*: This *.DLL* must be either in your path or in the same directory as the application you are writing.

For Visual Basic Developers

Include the following declaration at the top of the module:

- Declare Function SSTPrint Lib "SSTDEBUG" (ByVal DebuggerType as Integer, ByVal Message as String)

For Powerbuilder Developers

Include the following declaration at the top of the module:

- Function INTEGER SSTPrint (INTEGER DebuggerType,STRING MessageString) LIBRARY "SSTDEBUG"

SST Function Descriptions

The following functions are available in *SSTDEBUG.DLL*:

SSTPrintf()

BOOL FAR __cdecl SSTPrintf(WORD DebuggerType,LPSTR szFormat, ...);

Arguments

int DebuggerType

May be one of the following:

- **DEBUGGER_TYPE_WINSOCK**
- **DEBUGGER_TYPE_ODBC**
- **DEBUGGER_TYPE_SQL_SERVER**
- **DEBUGGER_TYPE_WINDOWS**
- **DEBUGGER_TYPE_ALL**

Note by OR'ing the **DebuggerType** flags together, you can send the same comment to multiple TracePlus applications. This feature is commonly used to create checkpoints while tracing multiple APIs, such as ODBC and Winsock.

char *szFormat

A null terminated string containing any valid **printf** style formatting string. See your C language reference for information on valid format specifications.

argument

One or more optional parameters that will be formatted according to the format specification.

Returns

No return value.

Comments

SSTPrint() will create an entry in the trace log:

- **1125**:   This is a comment.
- **12:35:00**:   Task: GENERIC

Sample Code

```
#include "sstdebug.h"

int PASCAL WinMain( HANDLE hInstance,HANDLE hPrevInstance,LPSTR lpCmdLine,int
nCmdShow )

{
MSG msg;                    /* message              */

SSTPrintf( "The value of hInstance is: 0x%X",hInstance );

while( TRUE ) {
      if ( PeekMessage( &msg,NULL,0,0,PM_REMOVE ) )
          if ( msg.message == WM_QUIT )
              break;
         else
               if ( !TranslateAccelerator( hMainWnd,hAccel,&msg ) ) {
                    TranslateMessage( &msg );
                    DispatchMessage( &msg );
                    }

      BackgroundProcessing();
      }

return( msg.wParam );       /* Returns the value from PostQuitMessage */
                    }
```

SSTPrintfEx()

BOOL FAR __cdecl SSTPrintfEx(WORD DebuggerType,LPSTR szFilename,int LineNumber,LPSTR szFormat, ...);

Arguments

int DebuggerType

May be one of the following:

- **DEBUGGER_TYPE_WINSOCK**
- **DEBUGGER_TYPE_ODBC**
- **DEBUGGER_TYPE_SQL_SERVER**
- **DEBUGGER_TYPE_WINDOWS**
- **DEBUGGER_TYPE_ALL**

Note by OR'ing the **DebuggerType** flags together, you can send the same comment to multiple TracePlus applications. This feature is commonly used to create checkpoints while tracing multiple APIs, such as ODBC and Winsock.

LPSTR szFilename

The filename of the source code where this function is being called.

int LineNumber

The source code line number where this function is being called.

char *szFormat

A null terminated string containing any valid **printf** style formatting string. See your C language reference for information on valid format specifications.

argument

One or more optional parameters that will be formatted according to the format specification.

Returns

No return value.

Comments

SSTPrintfEx() will create an entry in the trace log:

- **1125**:   This is a comment.
- **12:35:00**:   Task: GENERIC

Sample Code

```
#include "sstdebug.h"

int PASCAL WinMain( HANDLE hInstance,HANDLE hPrevInstance,LPSTR lpCmdLine,int
nCmdShow )

 {
MSG msg;                          /* message                 */

SSTPrintfEx( __FILE__,__LINE__,"The value of hInstance is: 0x%X",hInstance );

while( TRUE ) {
        if ( PeekMessage( &msg,NULL,0,0,PM_REMOVE ) )
                  if ( msg.message == WM_QUIT )
                       break;
```

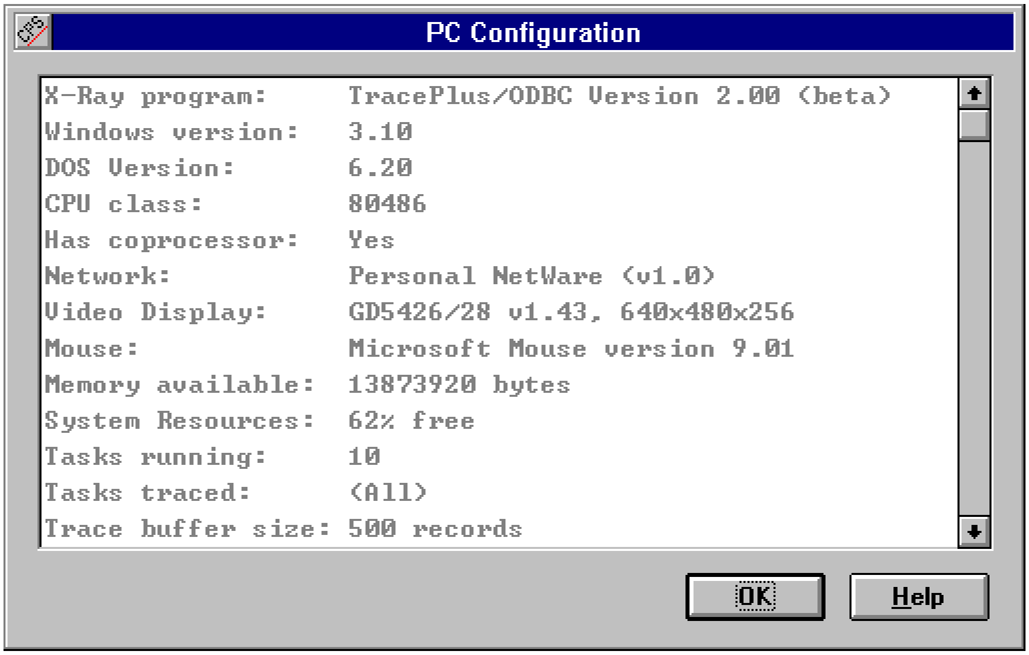
```
                    else
                          if ( !TranslateAccelerator( hMainWnd,hAccel,&msg ) ) {
                               TranslateMessage( &msg );
                               DispatchMessage( &msg );
                                }

          BackgroundProcessing();
          }
return( msg.wParam );          /* Returns the value from PostQuitMessage */
}
```

PC Configuration

To view the remote PC configuration from which a log file was exported/imported, select the PC Config button in the Import Trace Records dialog box. Alternatively, after you close the Import Trace Records dialog box, select PC Configuration... (View menu).

The PC Configuration dialog box is displayed.



```
X-Ray program:     TracePlus/ODBC Version 2.00 (beta)
Windows version:   3.10
DOS Version:       6.20
CPU class:         80486
Has coprocessor:   Yes
Network:           Personal NetWare (v1.0)
Video Display:     GD5426/28 v1.43, 640x480x256
Mouse:             Microsoft Mouse version 9.01
Memory available:  13873920 bytes
System Resources:  62% free
Tasks running:     10
Tasks traced:      (All)
Trace buffer size: 500 records
```

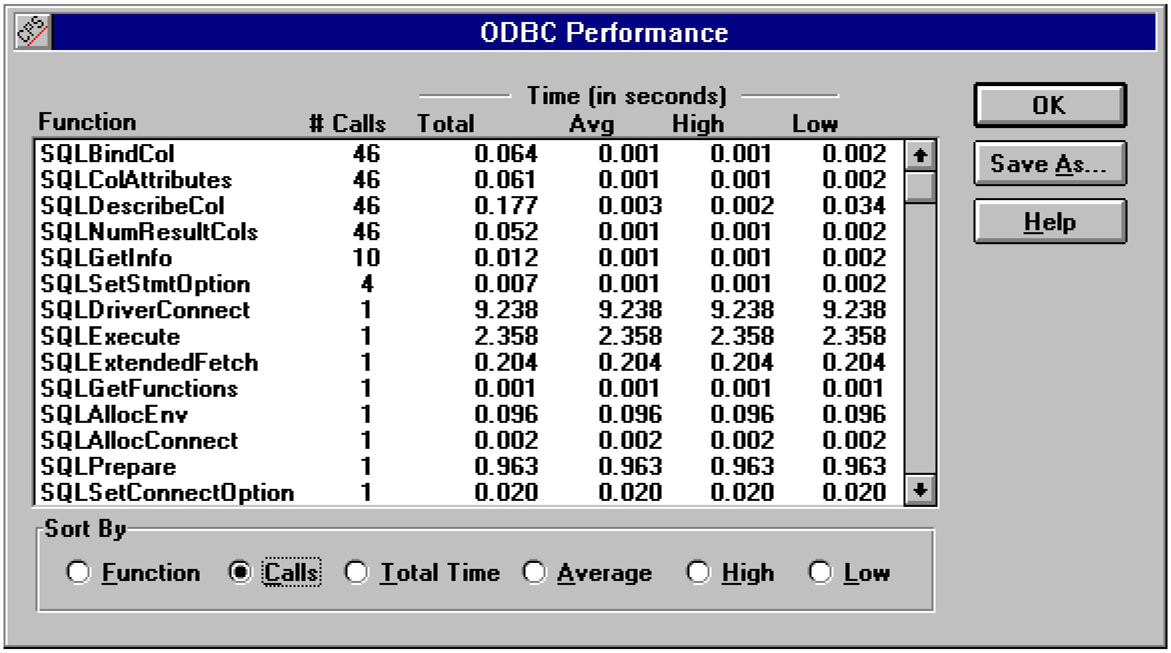
Viewing API Performance

The following TracePlus applications have the ability to record, display, and export performance information:

- TracePlus/ODBC
- TracePlus/SQL Server
- TracePlus/Winsock

Select API Performance... (View menu) to display the total calls, total time, average time per call, and the fastest and slowest recorded times for each call.

The following dialog box is displayed. Click on option fields and buttons for more information about how to use this dialog box.



ODBC Performance

		Time (in seconds)			
Function	# Calls	Total	Avg	High	Low
SQLBindCol	46	0.064	0.001	0.001	0.002
SQLColAttributes	46	0.061	0.001	0.001	0.002
SQLDescribeCol	46	0.177	0.003	0.002	0.034
SQLNumResultCols	46	0.052	0.001	0.001	0.002
SQLGetInfo	10	0.012	0.001	0.001	0.002
SQLSetStmtOption	4	0.007	0.001	0.001	0.002
SQLDriverConnect	1	9.238	9.238	9.238	9.238
SQLExecute	1	2.358	2.358	2.358	2.358
SQLExtendedFetch	1	0.204	0.204	0.204	0.204
SQLGetFunctions	1	0.001	0.001	0.001	0.001
SQLAllocEnv	1	0.096	0.096	0.096	0.096
SQLAllocConnect	1	0.002	0.002	0.002	0.002
SQLPrepare	1	0.963	0.963	0.963	0.963
SQLSetConnectOption	1	0.020	0.020	0.020	0.020

Sort By

○ Function ● Calls ○ Total Time ○ Average ○ High ○ Low

Launching the ODBC Administrator

Select ODBC Administrator (View menu).

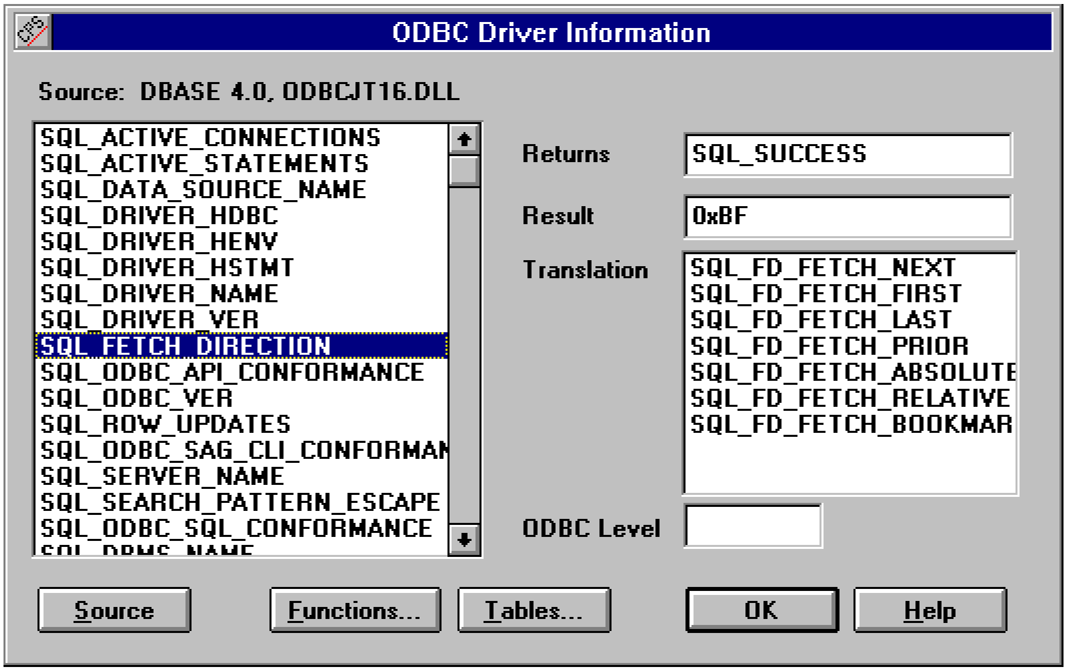
Starting SQL Capture

Select SQL Capture... or press **Ctrl** + **Q** to start the SQL Capture utility. SQL Capture records all SQL statements obtained during this tracing, as well as the elapsed time of the statement execution. You can export this information, in tab-delimited format, into Microsoft Excel or another spreadsheet for viewing.

Viewing Data Source Information

Select Data Source Information... (View menu) to view all available parameters for a selected ODBC data source.

The following dialog box is displayed. Select any of the buttons for more information on how to use this dialog box.



Source: DBASE 4.0, ODBCJT16.DLL

SQL_ACTIVE_CONNECTIONS
SQL_ACTIVE_STATEMENTS
SQL_DATA_SOURCE_NAME
SQL_DRIVER_HDBC
SQL_DRIVER_HENV
SQL_DRIVER_HSTMT
SQL_DRIVER_NAME
SQL_DRIVER_VER
SQL_FETCH_DIRECTION
SQL_ODBC_API_CONFORMANCE
SQL_ODBC_VER
SQL_ROW_UPDATES
SQL_ODBC_SAG_CLI_CONFORMAN
SQL_SERVER_NAME
SQL_SEARCH_PATTERN_ESCAPE
SQL_ODBC_SQL_CONFORMANCE
SQL_DBMS_NAME

Returns: SQL_SUCCESS

Result: 0xBF

Translation:
SQL_FD_FETCH_NEXT
SQL_FD_FETCH_FIRST
SQL_FD_FETCH_LAST
SQL_FD_FETCH_PRIOR
SQL_FD_FETCH_ABSOLUTE
SQL_FD_FETCH_RELATIVE
SQL_FD_FETCH_BOOKMAR

ODBC Level

Viewing Winsock Information

The Winsock Information dialog displays the following information about the Winsock implementation:

- Highest compatible Winsock version.
- Copyright notice returned from WSAStartup().
- Maximum open sockets per process.
- Maximum size of a UDP packet.
- All open sockets. The list is refreshed in realtime approximately every 10 seconds.

There is a caveat when using the *Open Sockets by Task* feature:

- Only the tasks selected in the Filters dialog box will be monitored for open sockets.

Note that sockets opened by an application that are not subsequently closed (i.e. the application UAEd, logic error, etc.) will remain in the open sockets list until the Delete button is pressed. At that time all sockets belonging to tasks that no longer exist will be removed from the list.

There are three states that will be displayed for an open socket:

1 OPEN
2 LISTEN
3 CONNECTED

Specifying the API Location

TracePlus/Winsock

This function is provided so that you can   use TracePlus/Winsock with multiple TCP/IP stacks. TracePlus loads the *WINSOCK.DLL* file you specify in the Trace Options dialog box.

TracePlus/SQL Server

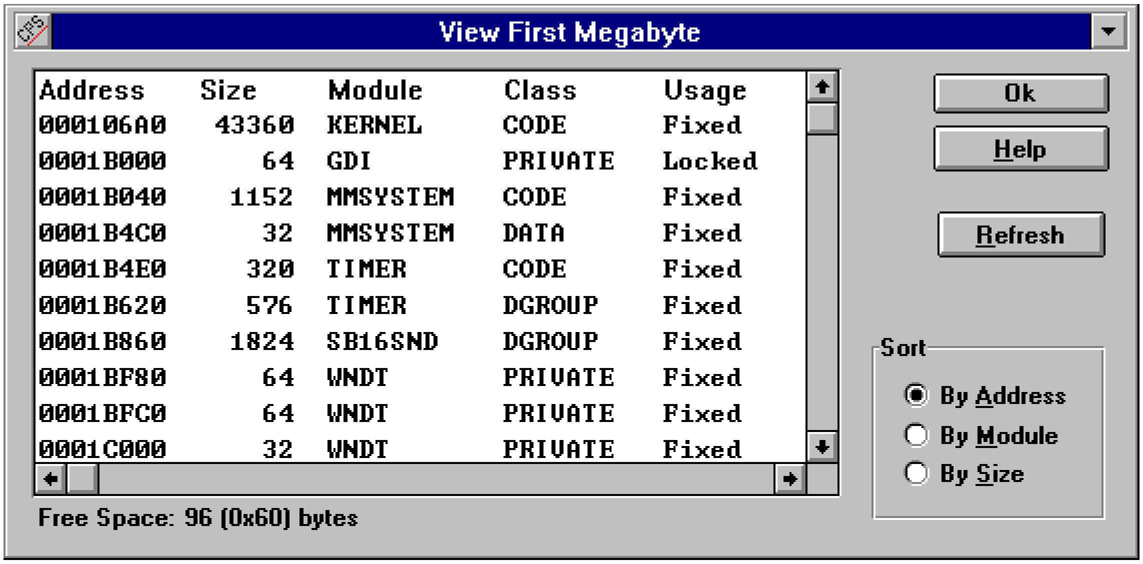
This function is provided so that you can use TracePlus/SQL Server with both Microsoft and Sybase versions of DB-Library. TracePlus/SQL Server defaults to using W3DBLIB.DLL. However, TracePlus will load the *DLL* you specify in the Trace Options dialog box.

Displaying Low Memory

Select First Megabyte... (View menu) to display the status of all memory blocks below the critical one megabyte boundary. Alternatively, press **Ctrl** + **V**.

Low memory is a scarce resource, and it is important to know how much low memory your application is using if you are using DDESHAREd or GlobalDOSAlloc'd memory.

The following dialog box is displayed. Click on the Sort field, or on the Refresh button for more information on how to use this dialog box.



```
Address     Size    Module     Class     Usage
000106A0   43360    KERNEL     CODE      Fixed
0001B000      64    GDI        PRIVATE   Locked
0001B040    1152    MMSYSTEM   CODE      Fixed
0001B4C0      32    MMSYSTEM   DATA      Fixed
0001B4E0     320    TIMER      CODE      Fixed
0001B620     576    TIMER      DGROUP    Fixed
0001B860    1824    SB16SND    DGROUP    Fixed
0001BF80      64    WNDT       PRIVATE   Fixed
0001BFC0      64    WNDT       PRIVATE   Fixed
0001C000      32    WNDT       PRIVATE   Fixed
```

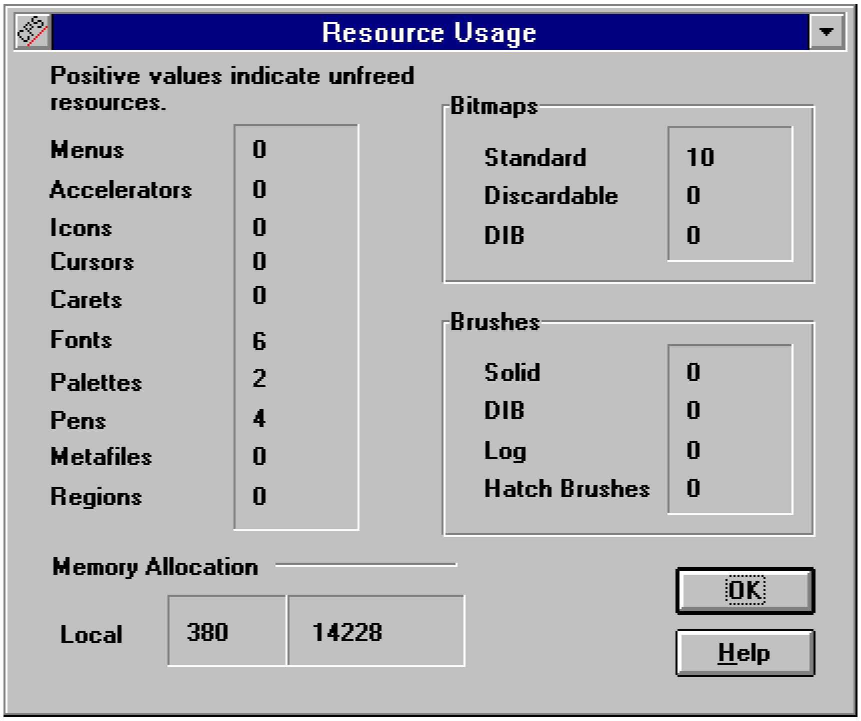
Free Space: 96 (0x60) bytes

Select Ok or press **Enter** to exit the dialog box.

Displaying Resource Usage

Select Resource Usage... (View menu) to display resource usage by GDI object. Alternatively, press **Ctrl** + **U**.

The following dialog box is displayed.



Positive values indicate unfreed resources.

Menus	0
Accelerators	0
Icons	0
Cursors	0
Carets	0
Fonts	6
Palettes	2
Pens	4
Metafiles	0
Regions	0

Bitmaps

Standard	10
Discardable	0
DIB	0

Brushes

Solid	0
DIB	0
Log	0
Hatch Brushes	0

Memory Allocation

Local	380	14228

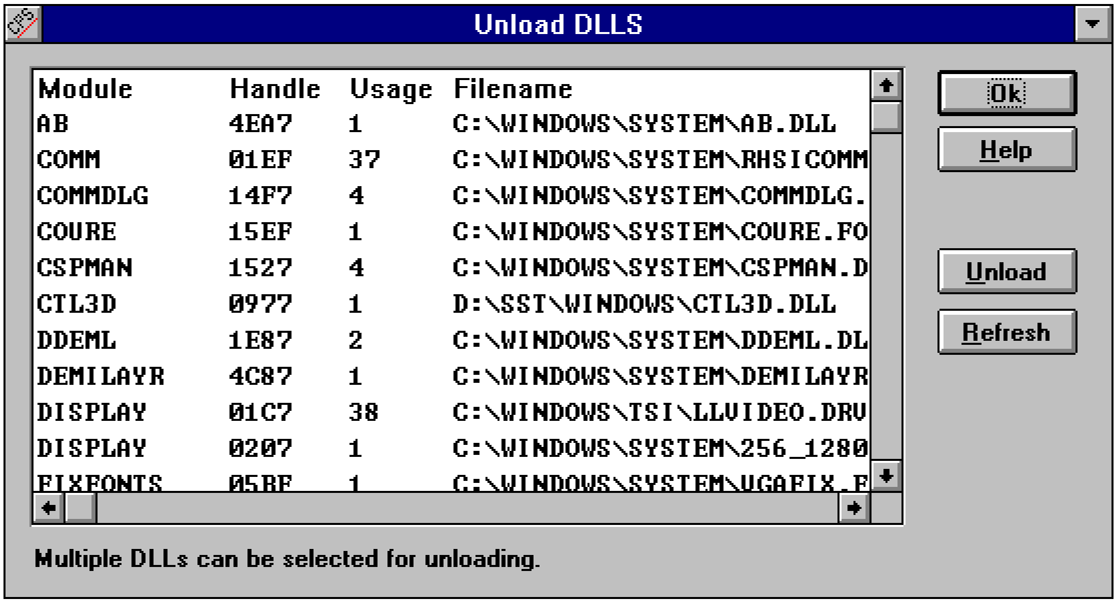
Each time an object is allocated, the count for that object is incremented by one. Each time an object is deallocated, the count for that object is decremented by one. If the count for any one object is above zero when your program terminates, that object has not been freed.

In addition, the local memory tracks current (not freed) allocations, and the total local memory allocated by the application.

Unloading DLLs

Select Unload DLLs... (File menu) to unload DLLs. Alternatively, press **Ctrl** + **D**.

The following dialog box is displayed. Click on the Ok, Refresh, or Unload button for more information on how to use this dialog box.



Module	Handle	Usage	Filename
AB	4EA7	1	C:\WINDOWS\SYSTEM\AB.DLL
COMM	01EF	37	C:\WINDOWS\SYSTEM\RHSICOMM
COMMDLG	14F7	4	C:\WINDOWS\SYSTEM\COMMDLG.
COURE	15EF	1	C:\WINDOWS\SYSTEM\COURE.FO
CSPMAN	1527	4	C:\WINDOWS\SYSTEM\CSPMAN.D
CTL3D	0977	1	D:\SST\WINDOWS\CTL3D.DLL
DDEML	1E87	2	C:\WINDOWS\SYSTEM\DDEML.DL
DEMILAYR	4C87	1	C:\WINDOWS\SYSTEM\DEMILAYR
DISPLAY	01C7	38	C:\WINDOWS\TSI\LLVIDEO.DRV
DISPLAY	0207	1	C:\WINDOWS\SYSTEM\256_1280
FIXFONTS	05BF	1	C:\WINDOWS\SYSTEM\VGAFIX.F

Multiple DLLs can be selected for unloading.

This dialog box displays all loaded modules, including DLLs (fully qualified for LoadLibrary ( )) and .DRV files, in the system.

Keyboard

Following are the TracePlus keyboard shortcuts.

Access API help on ODBC	Ctrl + F1
Access API help on SQL Server	Ctrl + F1
Access API help on Windows	Ctrl + F1
Access API help on Winsock	Ctrl + F1
Access help on specific API function	Shift + F1
Access TracePlus help	F1
Append a comment to a trace	Ctrl + Shift + A
Clear trace log and buffer	Ctrl + R
Continue the trace	Ctrl + C
Display event details	Ctrl + E
Display low memory (Windows)	Ctrl +V
Display resource usage (Windows)	Ctrl + U
Exit TracePlus	Alt + F4
Find a record	F3
Pause the trace	Ctrl + A
Print	Ctrl + P
Set trace functions and tasks	Ctrl + F
Set trace preferences	Ctrl + O
Start SQL Capture (ODBC/SQL Server)	Ctrl + Q
Start the trace	Ctrl + S
Unload DLLs (Windows)	Ctrl + D

Toolbar

The following command icons are available on the tool bar:

Saves the contents of the current trace log as an ASCII file.

Prints the contents of the screen.

Displays the details of a specified event.

Opens the Filters dialog box, so that you can select the functions and tasks (applications) you want to trace.

Opens the Trace Options dialog box, so that you can
set trace preferences.

Allows you to append a comment to the current trace.

Clears the trace log and buffer.

Starts trace, and clears the current trace

Stops tracing events.

Continue tracing events.   It saves the current trace.

Displays the API help file for ODBC, SQL Server, Windows, Winsock, or WLibsock.

Displays help on a specific API function.

Index

≜
#
A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z

API Functions

Select the trace functions you want to run.

For more information about ODBC functions, refer to the documentation that accompanies the ODBC API.   TracePlus does not trace the ODBC function **SQLGetData()**.

For more information about SQL Server functions, refer to the documentation that accompanies the SQL Server API. TracePlus does not trace **dbreadpage**, **dbwritepage**, and **dbfcmd**.

For more information about these functions, refer to the documentation that accompanies the Winsock API.   TracePlus does not trace the Winsock function **WSAGetLastError()**.

Backwards

Select this option if you want to see the log in reverse order while animation is enabled.

Browse Records

Select the Browse scroll buttons to view records ahead or behind the current record.

Buffer Options Field

The number in parentheses below the Buffer Size option is the amount of memory in kilobytes.

Buffer Size: Enter the number of records, between 25-2000 inclusive, that you want to buffer.

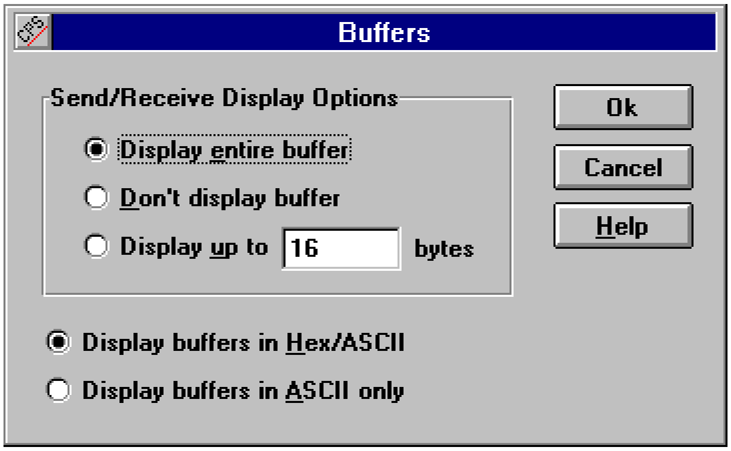
Circular: When the number of records is one more than the number you specified in the **Buffer Size** option, the buffer drops the oldest record, giving the appearance of an endless buffer. The larger the buffer size, the more records it can hold before losing the oldest record.

Stop when filled: The buffer stops accepting record when the number of records you specified in the **Buffer Size** option is reached. The trace is automatically stopped.

Buffers

Select this button to specify the data buffers display in the Event Details dialog box.   (Typically buffers passed to **recv()**/**recvfrom()** or obtained from **send()**/**sendto()**.)

Select any field or button below for more information on using this dialog box.



● Display entire buffer

○ Don't display buffer

○ Display up to 16 bytes

● Display buffers in Hex/ASCII

○ Display buffers in ASCII only

TracePlus will consume much more memory if you are tracing applications that send and receive data in large packet sizes (2048, 4096, etc.). To minimize memory usage, set a limit for buffer sizes, or turn off the buffer display altogether. Alternatively, you can set the trace buffer size to a smaller value, typically one half of the current buffer size.

Call

Select this button to display a description of the function shown in the details list box as outlined in the API help file. This button is grayed if the API help file is not accessible by TracePlus.

Cancel

Select this button to exit the dialog box without making any changes.

Choose Application

To select the application, select the Choose application... button, or enter the application in the **Application** text box.   You can enter arguments to the application in the **Command Line** text box.

Clear

Select this button to cancel your choice(s) in the list box.

Close

Select this button to close the dialog box.

DBWin

Outputs trace information to DBWin (a sample debugger available with Microsoft Windows SDK and Microsoft C/C++ (Visual C++TM)). This option also supports output to WinScopeTM by Periscope. TracePlus is compatible with any program capable of displaying messages sent by the function **OutputDebugString( )**.

Be aware that this option can slow your system down if many trace events are being logged.

Detail List Box

This window displays detail of a specified trace. You can specify the trace for which you want more detail by double clicking on a trace record in the TracePlus window, or by entering a number in the Record text box and pressing the Get button. Alternatively, you can scroll through the list box, or select the Animate button to view other records, or select the Find button to find a record alphanumberically.

The following information is displayed for every trace detail:

- **FUNCTION SUMMARY**: A summary of the function, including record number, function, and the time it was called.
- **PARAMETERS BEFORE CALL**: Parameters in the code before the function is executed.
- **RETURNS**: Error codes or values that are returned by the function.
- **PARAMETERS AFTER CALL**: Values that may have been changed by the function.

TracePlus/VBX+Windows additions

- **VERSION INFORMATION**: This section is displayed when a LoadLibrary() trace record is generated. It contains the information received by calling the Microsoft Version API, and includes version information, copyright and trademark notices, target OS, language type,
- **VALID PROPERTY SETTINGS**: If the VBX control property is an enumerated property, TracePlus will display the the list of valid settings for that property. Currently, this does not apply to standard properties.
- **DATA ACCESS STRUCTURE**: This is the Microsoft defined "C" structure that sent between the bound control and the data control, specifying the action to take by the receiving control.
- **VISUAL BASIC DECLARATION**: This is the predefined syntax for a VBX event procedure, as specified by the VBX control.
- **NEW PROPERTY VALUE**: The value that will be assigned to a property (displayed when a VBSetControlProperty() trace record is generated).
- **VB MESSAGE DETAILS**: The values of each parameter passed to the VBX control procedure, corresponding to the standard Windows message format, i.e. (HWND hWnd,unsigned msg,WPARAM wParam,LPARAM lParam).
- **VBX EVENT SUMMARY**: The parameter values passed to the VBFireEvent() API function.

Display buffers in ASCII only

This option displays buffer in text format.

Display buffers in Hex ASCII

This option displays buffer information in DEBUG style format.

Display Options Field

Minimize when trace started: Minimizes the TracePlus application window to an icon.

Stop trace when activated: Stops the trace when you activate the TracePlus window.

Always on top: Keeps the TracePlus window on top of all other applications, so that it is always visible.

Enable VBX Tracing

This option determines whether or not TracePlus/VBX will intercept the activity between your application and the selected VBX controls.

Errors

Select the Errors scroll buttons to scroll through errors.

Errors display result codes

If you are using TracePlus/Winsock, this checkbox selects whether to display result codes or text when Winsock errors occur.

If you are using TracePlus/ODBC or TracePlus/SQL Server, this checkbox is instead labeled *Display Results after Fetch*. When this feature is enabled, TracePlus saves the locations of all columns bound to the SQL Statement just executed, and will append the contents of these columns to the trace record.

The trace information is altered for the following ODBC functions:

- SQLFetch()
- SQLExtendedFetch()

The trace information is enhanced for the following DB-Library functions:

- dbnextrow()

Errors Only

Select this option if you want to view error messages while animation is enabled.

File

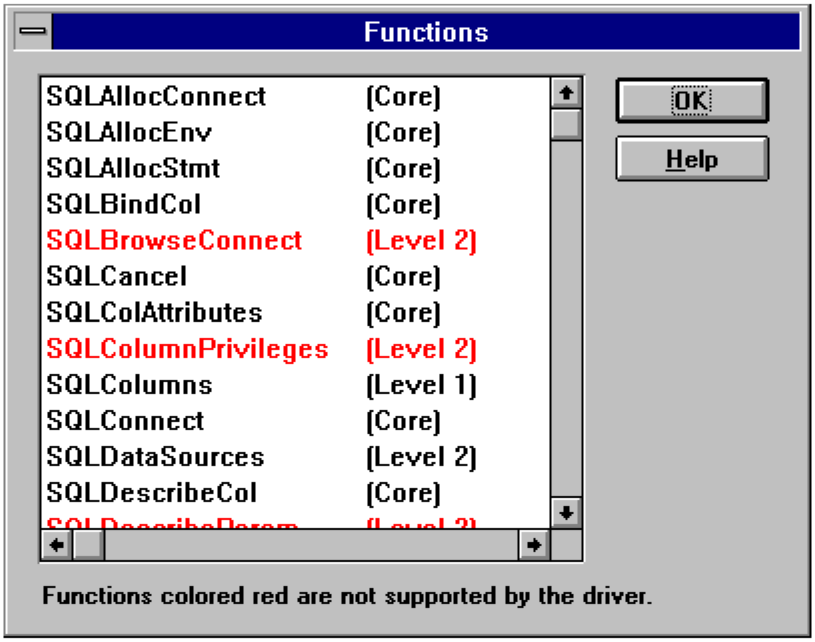
Outputs trace information to a log file.   The default log file for TracePlus/ODBC is *C:\ODBC.LOG*. The default log file for TracePlus/SQL Server is *C:\ODBC.LOG*. The default log file for TracePlus/SQL Server is *C:\SQLSERVR.LOG*. However, you can enter another file in the Log File text box, or choose another log file by selecting the Browse button.

The default is **Append to existing file**, which appends new trace information to the log file you specify.   If you deselect the default, the log file is deleted and a new file is created when you run a trace on another function.

Note that if you do not specify the log file, you can save the contents of a trace buffer in ASCII format to a file by selecting Save Trace As... (File menu).

Functions

Select this button to view the ODBC functions supported by the ODBC driver for the selected data source.



SQLAllocConnect	[Core]
SQLAllocEnv	[Core]
SQLAllocStmt	[Core]
SQLBindCol	[Core]
SQLBrowseConnect	[Level 2]
SQLCancel	[Core]
SQLColAttributes	[Core]
SQLColumnPrivileges	[Level 2]
SQLColumns	[Level 1]
SQLConnect	[Core]
SQLDataSources	[Level 2]
SQLDescribeCol	[Core]
SQLDescribeParam	[Level 2]

Functions colored red are not supported by the driver.

Supported functions are in black.   Unsupported functions are in red.

Select Ok or press **Enter** to exit the dialog box and return to the Data Source Information dialog box.

Get

Select this button to retrieve details for a trace you specify in the Record text box. The record is the first field that is displayed in the TracePlus window. You can also use the scroll buttons to scroll backwards and forwards through record numbers.

Include Types

Select **System**, **Views**, or **Aliases** for the type of column you want to view.

Select Ok or press **Enter** to exit the dialog box and return to the ODBC Driver Information dialog box.

Launch specified application

Select this option if you want to run a trace on an application that is not displayed in the Tasks list box.   To select the application, select the Choose application... button, or enter the application in the **Application** text box.   You can enter arguments to the application in the **Command Line** text box.

Level of detail

Select **High** to output information details to printer, DBWin, or the file you specified in the option above. Select **Low** to display only two lines of information to the Event Details dialog box.

Remember that this option does not change the format of the TracePlus main window. This options affects either the **File** or **Printer** output options.

Log Exceptions Only

Select this option if you want to log functions that return errors only.

Log File

If you selected **File** in the Output Options field, and you do not want to accept the default log file, enter the name of the file to which you want trace information stored, or choose another log file by selecting the Browse button.

The default log file for TracePlus/Winsock is *C:\WINSOCK.LOG*. The default log file for TracePlus/Wlibsock is *C:\WLIBSOCK.LOG*. The default log file for TracePlus/ODBC is **C:\ODBC.LOG**. The default log file for TracePlus/SQL Server is *C:\SQLSERVR.LOG*.

Note that if you do not specify the log file, you can save the contents a trace buffer in ASCII format to a file by selecting Save Trace As... (File menu).

Machine name

The name of the workstation is automatically displayed in this field under Windows NT and Windows for Workgroups. This differentiates users when multiple workstations are traced by writing to the same log file on a network drive.

Menus

Insert Menus help text here

Monochrome Display

For a monochrome monitor.   This option is grayed if you do not have a monochrome video card installed in your computer.

Ok

Select the Ok button or press **Enter** to set your changes and exit the dialog box.

Printer

Outputs trace information to your printer.

Property Messages

When an application or a VBX control calls the API function *VBSetControlProperty*(), a VBM_SETPROPERTY message may be generated, if the property was not created with the PF_fSetData flag. Therefore, if you have not filtered standard and custom properties, the VBSetControlProperty() trace record will appear as follows:

1 Control1.Property = "Value" (This is the VBM_SETPROPERTY message)

2 Control1.Property = "Value" (This is the VBSetControlProperty() API call.

This same behavior occurs when the API function VBGetControlProperty() is called.

Refresh

Refresh the list box.

Save As

Select this button to save the performance statistics as a text file.

Select All

To select individual list box items, press **Ctrl** as you click with the left mouse button. To select a range of list box items, click on the first item with the left mouse button, and then press **Shift** as you click on the last item with the left mouse button. To select all items in the list box, select the Select All button.

Select records to load

Select All to load all records, or specify the record number at which you want to start. Note that the total number of records is specified for your convenience.

Send Receive Display Options

Display entire buffer: Displays the entire buffer in the selected format.

Don't display buffer: Displays "N/A" instead of the contents of the buffer.

Display up to nnnn bytes: Displays the first nnnn bytes of the buffer. Buffers are limited to 4096 bytes. If a buffer is larger than 4096 bytes, it will be truncated to 4096.

Skip or Get

If you select the **Skip if WSAWOULDBLOCK** option for TracePlus/Winsock, WSAWOULDBLOCK is not logged. WSAWOULDBLOCK is returned if the **recv()**, **recvfrom()**, **send()**, and/or **sendto()** functions fail because no data could be received or sent.

If you select the **Get extended error information** option for TracePlus/ODBC, the ODBC function **SQLError()** returns verbose error information that is stored in the **henv**, **hdbc**, and **hstmt** structures when an error occurs. ODBC returns a string containing information on each error that has occurred. Also, ODBC can generate multiple error statements from one function call. TracePlus will display these strings when an API function returns an error. TracePlus will also display the error codes (known as native error codes) specific to the type of database you are using. By cross-referencing these error codes with the database vendor, these codes may shed light on some more less obvious problems you may be having. If this option is not selected, the generic return values will displayed.

For TracePlus/SQL Server, select **Get extended error information** if you want to utilize the error handler via **dberrhandle()**. This function reports four pieces of information back to the application that called the function:

- **dberr** - The identifying number of the error
- **oser** - The error number returned by the operating system or network
- **dberrstr** - Verbose information about **dberr**
- **oserrstr** - Verbose information about **oserr**

TracePlus/SQL Server displays the values of these items when an API function returns an error. If this option is not selected, the generic return value will displayed.

For TracePlus/Winsock, select **Skip if WSAWOULDBLOCK** if you do not want EWOULDBLOCK logged. WSAWOULDBLOCK is returned if the **recv**, **recvfrom**, **send**, and/or **sendto** functions fail because no data could be received or sent.

For TracePlus/WLibsock, select **Skip if EWOULDBLOCK** if you do not want EWOULDBLOCK logged. EWOULDBLOCK is returned if the **recv**, **recvfrom**, **send**, and/or **sendto** functions fail because no data could be received or sent.

Sort

Select the option by which you want to sort information

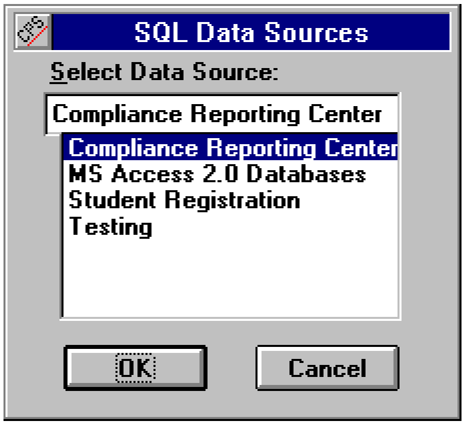
- If you choose address, addresses are listed in ascending order.
- Module refers to the module that allocates the memory.
- Size refers to the amount of memory the block occupies.

Sort By

Select the sort option by which you want to display performance statistics.

Source

When the ODBC Driver Information dialog box is displayed, the SQL Data Sources dialog box is also automatically displayed to allow you to select the data source.   You can also select the Source button to display the SQL Data Sources dialog box.



Select Data Source: Compliance Reporting Center

Select the data source and select Ok or press **Enter**.

Speed

Enter the number from 1 to 15. The higher the number, the slower the animation (that is, automated scrolling through the detail list box).   The range of values equates to .2 through 3 seconds. You can initiate animation by selecting the Start key.

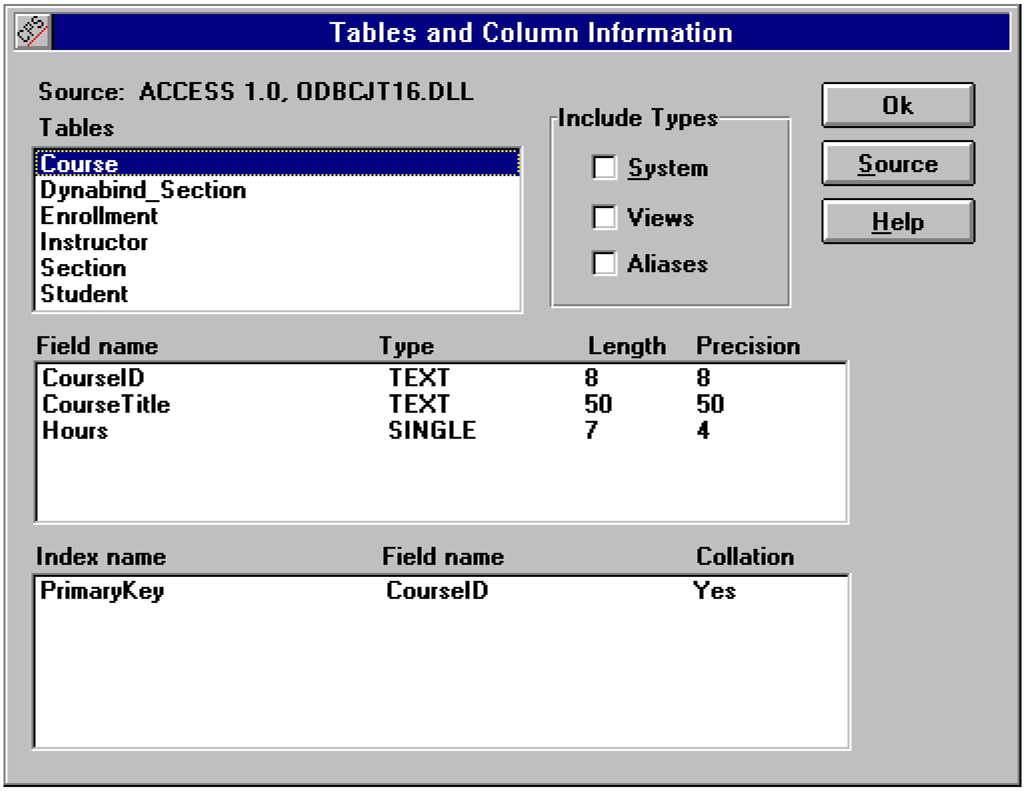
Start

Select this button if you want to automate scrolling through the detail list box. To set the animation delay, enter the number of seconds delay in the seconds text box.

Tables

Select this button to view the columns and indexes for each table within the database for the data source you selected in the SQL Data Sources dialog box.

The following dialog box is displayed.   Select field options and buttons for more information about how to use this dialog box.



Source: ACCESS 1.0, ODBCJT16.DLL

Tables

- Course
- Dynabind_Section
- Enrollment
- Instructor
- Section
- Student

Include Types

□ System

□ Views

□ Aliases

Field name	Type	Length	Precision
CourseID	TEXT	8	8
CourseTitle	TEXT	50	50
Hours	SINGLE	7	4

Index name	Field name	Collation
PrimaryKey	CourseID	Yes

Tasks

Select the applications for which you want trace information.

Trace all running applications

Select this option if you want to trace all current and new applications.

- Note that this option is only available in 16 bit versions of TracePlus.

32 bit versions of TracePlus have an option called *Trace Processes started from Desktop*. This feature will automatically trace an application under two conditions:

- The application has a main window (i.e., not a Win32 console application).
- The trace will be initiated at the point where the main window receives the WM_CREATE message. This means that any API calls made before this message is received will not be traced.

The procedure for tracing 32 bit applications launched from the desktop is as follows:

1. Make sure that a checkmark is next to the item *Trace processes started from desktop* in the Filters dialog box.
2. Unselect the option *Launch specified application*
3. Press OK (not Cancel) to close the **Filters** dialog box.
4. Click on the *Start* button on the toolbar, or press Ctrl+S.
5. Launch the Win32 application that you want to trace. Make sure that it creates a main window.

Trace log only

Displays trace information to a window.

Unload

Select the module(s) you want to unload and then select this button. Note that you can unload multiple modules; this is especially handy if you are developing ODBC applications, which leave multiple DLLs stuck in memory if the application GPF's.

VBX Filters

These settings reduce the number of VBX trace records by ignoring the specified VBX actions. Each filter applies globally to all selected VBXs.

Filters

- **Include VBX messages**: If this option is selected, all VBX messsages sent to the VBX control procedure will be displayed.
- **Include VBX events**: If this option is selected, all events fired by a VBX control will cause a trace record to be generated.
- **Standard properties**: If this option is selected, all standard property accesses (those properties predefined by Visual Basic) will appear in the trace log.
- **Custom properties**: If this option is selected, all accesses to properties defined by the maker of the VBX control will appear in the trace log.
- **Windows messages**: If this option is selected, any Windows (not VB) message that is sent to the control procedure of a VBX control will appear in the trace log.
- **Bound Control messages**: If this option is selected, any VBM_DATA_* messages sent between bound controls and the data control will appear in the trace log.

VBX List

When you start TracePlus/VBX+Windows, TracePlus creates a database of VBXs it found on your PC. The search criteria that TracePlus uses is:

1 The Windows system directory
2 The directories found in the DOS PATH statement.
3 The directory of the application you specified to launch via the **Filters** dialog box.

Each VBX found is scanned for VBX control definitions. A VBX file can have multiple control definitions. TracePlus also retrieves the version of Visual Basic supported by the VBX control.