SQL*Plus

Accessibility Guide

Release 8.1.7 for Windows

September 2000

Part No. A85448-01

About This Guide

This Accessibility Guide discusses the options available to SQL*Plus users who have special accessibility requirements, for example, users who want to use SQL*Plus with a screen reader.

Where possible, this document discusses third party accessibility products directly. For more information, or information on using a specific accessibility product, contact the provider of that product.

This Accessibility Guide discusses the following topics:

- Accessibility Background

- Accessibility and Oracle Products

- Oracle Partnerships

- Accessibility in SQL*Plus

    - Using Command Line SQL*Plus with JAWS

    - Using SQL*Plus for Windows with JAWS

ORACLE®

Accessibility Background

Rapid technical changes in software development have created sophisticated applications and tools designed to meet the challenges of changing marketplace requirements for businesses. The explosion of the Internet and E-commerce have required many businesses to re-invent themselves in order to remain competitive, and a plethora of products, web sites and E-commerce services have emerged. But with rapid technological changes can also come additional challenges for disabled users who may be confronted with obstacles that inadvertently block access to products and services.

Historically, the disabled user was provided with a customized solution. But given the E-commerce solutions with built-in color keys, drop down menus, GUI interfaces that require a mouse, PDF plug-ins that are not readable by screen readers, custom solutions for individual disabled users in an enterprise will not work. Accessibility needs to be part of the basic design so that enterprise wide solutions can accommodate the changing needs of users.

Accessibility is an issue of significance for a growing user population as it includes not only those born disabled, but those who have experienced incapacitation through accident, illness or aging. While the totally blind technical user population may seem to be a small population, the low vision or color blind user population is substantially larger. Hearing impairments, motor or cognitive disorders affect millions. As the overall incidence of disability increases with age, access to technology is a matter that concerns everyone.

Oracle's goal is to ensure that products and services are accessible to the disabled community with good usability. Industry standards will continue to evolve over time and Oracle is actively engaged with other market leading technology vendors in addressing apparent technical hurdles.

Accessibility and Oracle Products

The Accessibility Design Standards state that technology products like those produced by Oracle should provide universal access with a screen reader connected to a GUI and full keyboard access, or substantially equivalent access through a command line or textual method of performing the same tasks without a mouse or GUI.

Consistent with design standards pertaining to auditory product features, full product accessibility for hearing impaired users has also become part of Oracle product design goals. Oracle products incorporate the current standards for accessible design for the vision impaired who may be color sensitive, have low vision, or use a screen reader.

Oracle's accessibility initiative began in 1999 and changes will continue to evolve over time, that is, changes in both standards and product development. While accessibility may not be a new issue, industry standards upon which information technology products are developed are new.

Accessibility issues are global and affect everyone. Laws in Canada, Europe, Asia Pacific, and Japan are in development or are already in place and may become more rigorous. Participation in the Internet economy, removal of barriers to employment, and the shortage of skilled technical talent all contribute to making accessibility good business for people within the industry as well as technology users. Oracle firmly believes that advancements in information technology should benefit and be accessible to everyone.

Inquiries about Oracle's accessibility program can be directed to: accessible@us.oracle.com.

Inquiries about Oracle's accessibility program which are specifically related to SQL*Plus can be directed to: sqlplus@us.oracle.com.

Oracle Partnerships

Henter-Joyce, the market leading screen reader technology vendor, has collaborated with Oracle on JAWS. JAWS has become the office standard in many US Government agencies, is available in nine languages and has a global install base. For more information about JAWS, visit the Henter-Joyce web site at http://www.hj.com/.

Accessibility in SQL*Plus

The command line (DOS) version of SQL*Plus for Windows provides equivalent access, as defined in the Americans with Disabilities Act, 1990. Screen readers will have no difficulty in interpreting both the input and output of the command line version of SQL*Plus for Windows.

Note: Although screen readers may also read the output from the Windows version of SQL*Plus for Windows, Oracle recommends using the command line version of SQL*Plus for Windows with screen readers.

Using Command Line SQL*Plus with JAWS

The command line version of SQL*Plus for Windows has been tested with the JAWS screen reader version 3.5.

The default installation of JAWS is not configured to read the output of Microsoft Windows DOS windows. You can configure JAWS to read the output from DOS windows by changing the settings in the JAWS default

script DEFAULT.JSS. In a default JAWS installation, this script is located in C:\JFW35\settings\enu.

Configuring JAWS   To change the JAWS configuration to read the output of Microsoft Windows DOS windows, follow these steps:

1. Start the JAWS Script Manager program.

2. Open the file DEFAULT.JSS from the JAWS default installation directory.

3. Delete the C-style comment delimiters, /* and */, in the function SayNonHighlightedText. The original version of this function is:

```
Void Function SayNonHighlightedText (handle hwnd, string buffer)
; NonHighlightedText  Function for speaking all newly written
nonhighlighted
; text.
; Commented code added 09/24/99 by DB
; for use with Dos Boxes and Console apps
var
string TheClass
let TheClass = GetWindowClass(hwnd)
; for Windows 95, 98 and/or NT
; Just remove the c-style /* */ comments around this block of code
; if you wish Jaws to speak automatically in Dos Windows
; for text output, such as command prompt or command-line apps such
as FTP and Ping
/*
If (TheClass == "tty" || TheClass == "ttygrab" || TheClass ==
"ConsoleWindowClass") then
Say (buffer, OT_BUFFER);
Return
EndIf
*/
;wn36="#32771"
if ((GetScreenEcho() > 1) || (TheClass == wn36 )) then Say (buffer,
ot_buffer) EndIf
EndFunction
```

After removing the C-style comments, the function should be the same as below:

```
Void Function SayNonHighlightedText (handle hwnd, string buffer)
; NonHighlightedText  Function for speaking all newly written
nonhighlighted
; text.
; Commented code added 09/24/99 by DB
; for use with Dos Boxes and Console apps
var
```

```
string TheClass
let TheClass = GetWindowClass(hwnd)
; for Windows 95, 98 and/or NT
; Just remove the c-style /* */ comments around this block of code
; if you wish Jaws to speak automatically in Dos Windows
; for text output, such as command prompt or command-line apps such
as FTP and Ping
If (TheClass == "tty" || TheClass == "ttygrab" || TheClass ==
"ConsoleWindowClass") then
Say (buffer, OT_BUFFER);
Return
EndIf
;wn36="#32771"
if ((GetScreenEcho() > 1) || (TheClass == wn36 )) then Say (buffer,
ot_buffer) EndIf
EndFunction
```

4. Save DEFAULT.JSS and exit JAWS Script Manager. Note that JAWS compiles the script and makes updates to other files in the same directory.

Configuring JAWS Dictionary File   You can add entries into the JAWS dictionary file to improve the JAWS output from SQL*Plus. The following entries may be added to your DEFAULT.JDF file in the C:\JFW35\settings\enu directory.

```
.appinfo.app info.
.authid.auth i d.
.autocommit.auto commit
.autoprint.auto print.
.autorecovery.auto recovery.
.autotrace.auto trace.
.binary_integer.binary underscore integer.
.bindvariable.bind variable.
.block_terminator.block underscore terminator.
.btitle.b title.
.cmdsep.cmd sep.
.colsep.col sep.
.controlfile.control file.
.copycommit.copy commit.
.copytypecheck.copy type check.
.currval.curr val.
.datafile.data file.
.datafilename.data file name.
.dbname.db name.
.editfile.edit file.
.entmap.ent map.
.forall.for all.
.goto.go to.
```

```
.headsep.head sep.
.linenum.line number.
.linesize.line size.
.lno.l n o.
.loboffset.lob off set.
.logfile.log file.
.logsource.log source.
.longchunksize.long chunk size.
.maxextents.max extents.
.mlslabel.m l s label.
.naturaln.natural n.
.nchar.n char.
.nclob.n clob.
.newline.new line.
.newpage.new page.
.nextval.next val.
.noaudit.no audit.
.nocompress.no compress.
.nocopy.no copy.
.noduplicates.no duplicates.
.nolog.no log.
.nomount.no mount.
.noparallel.no parallel.
.noprint.no print.
.noprompt.no prompt.
.nowait.no wait.
.numformat.num format.
.numwidth.num width.
.nvarchar2.n var char 2.
.ocirowid.o c i row i d.
.offline.off line.
.oserror.o s error.
.pagesize.page size.
.pctfree.p c t free.
.pfile.p file.
.pno.p n o.
.positiven.positive n.
.preformat.pre format.
.printspec.print spec.
.recsep.rec sep.
.recsepchar.rec sep char.
.refcursor.ref cursor.
.repfooter.rep footer.
.repheader.rep header.
.rowid.row i d.
.rowlabel.row label.
.rownum.row num.
.rowtype.row type.
.savepoint.save point.
.sepchar.sep char.
```

```
.serveroutput.server output.
.shiftinout.shift in out.
.showmode.show mode.
.sql*plus.sequel plus.
.SQL*Plus.sequel plus.
.sql.sequel.
.sqlblanklines.sequel blank lines.
.sqlcase.sequel case.
.sqlcode.sequel code.
.sqlcontinue.sequel continue.
.sqlerrm.sequel err m.
.sqlerror.sequel error.
.sqlnumber.sequel number.
.sqlplus.sequel plus.
.sqlprefix.sequel prefix.
.sqlprompt.sequel prompt.
.sqlterminator.sequel terminator.
.stddev.s t d dev.
.sysdate.sis date.
.sysdba.sis d b A.
.sysoper.sis oper.
.termout.term out.
.timeout.time out.
.trimout.trim out.
.trimspool.trim spool.
.ttitle.t title.
.undefine.un define.
.username.user name.
.varchar.var char.
.whenever.when ever.
```

Note: This list is case sensitive. If you want JAWS to also read these SQL*Plus commands when they are in upper case, you must create another version of each command in upper case.

Using SQL*Plus for Windows with JAWS

The Windows version of SQL*Plus may also be used with JAWS. To achieve the best results, your screen resolution should be set to 800 by 600 pixels.

Configuring JAWS   To use the Windows version of SQL*Plus with JAWS, you must create two files and save them into the JAWS settings directory.

1.   Create a JAWS script using a text editor which contains the following:

```
:frame frm_response
:Synopsis Oracle response frame
:Description say Oracle's response
```

2. Save the file as SQLPLUSW.JFD in the C:\JFW35\settings\enu directory.

3. Create a second file which contains the following text:

```
[frm_response]
Echo=2
Prompt=
WindowBased=1
RelativeTo=0
OriginX=1
OriginY=41
Height=506
Width=42
RealWindowLeft=0
RealWindowTop=0
RealWindowRight=800
RealWindowBottom=577
NewTextScript=
FocusScript=
Priority=5
RealWindowName=Oracle SQL*Plus
RealWindowClass=SqlplusWClass
RealWindowText=
FrameText=
```

4. Save the file as SQLPLUSW.JFF in the C:\JFW35\settings\enu directory.

5. Start SQL*Plus in maximized mode, or maximize it after you open it.

You should also set JAWS user options for SQL*Plus for Windows:

1. Start JAWS.

2. Start SQL*Plus for Windows.

3. Press Insert + F2 to display the JAWS Configuration Manager.

4. Select Configuration Manager. The file SQLPLUSW.JCF should now be open.

5. Select User Options from the Set Options menu.

6. Select Words from the Typing Options radio buttons.

7. Select Echo All Text from Screen Echo radio buttons.

8. Select the Okay button.

9. Save SQLPLUSW.JCF.

10. Exit the JAWS Configuration Manager.