

BS2000/OSD Job Variables

The BS2000/OSD job variables facility provides functions that include LIST, EDIT, RENAME and DELETE job variables. You can also display information on a specific job variable, including its attributes and (if you are authorized to read it) its value. You can create a new job variable with standard attributes simply by editing and saving it, and you can modify the attributes of a job variable using the function command CHANGE.

All functionality described in this section is available only if SNI's software product JV has been installed as part of your BS2000/OSD system environment.

▶ To enter the BS2000/OSD job variable maintenance facility

- Select the JOB-VARS option from the Natural ISPF Main Menu.

The BS2000/OSD Job Variables Entry Panel appears:

```

----- BS2000 JOB VARIABLES - ENTRY PANEL -----
COMMAND ===>

      Job variable  ===> *

      Password      ===>                               ( If password protected )
      Node          ===> 31
  
```

You can specify the name of the intended job variable in the input fields and enter a function command in the command line.

Meaning of the input fields:

Field	Meaning
Job variable	Displays the job variable name last used. Select any other job variable by overtyping this name. Leave blank or use strings and wildcards (* and _) to generate a more selective list of variables. See the subsection Selection Windows and Wildcards in the section Command Logic.
Password	System password if job variable is protected. Must be specified irrespective of read or write protection.
Node	Select Entire System Server node. Enter a question mark ? and press Enter to open a window in which all node numbers are scrolled with an ACTIVE or INACTIVE status report. If you do not specify a node, the default node on the main menu is assumed.

Function Commands

The available function commands are:

Command	Parameter Syntax
CHANGE	jvar-name PASSWORD=p NODE=id, SHARE=YES/NO RETPD=nnnn ACCESS=READ/WRITE READPSWD=rpwd WRITEPSWD=wpwd
DELETE	jvar-name PASSWORD=p NODE=id
EDIT	jvar-name PASSWORD=p NODE=id
INFORM	jvar-name PASSWORD=p NODE=id
LIST	*_* NODE=id
RENAME	jvar-name PASSWORD=p NODE=id

Note:

If you issue any of the above function commands from outside the job variables facility, you must specify the object-type parameter JV before the object parameters.

A full description of these commands is contained in the section Command Reference. The object parameters correspond to the input fields on the BS2000/OSD Job Variables Entry Panel.

Listing Job Variables Selected by a Path Name Pattern

The specification can optionally contain a catalog ID and a different user ID. If so, only shared job variables are listed. If catalog ID and user ID are omitted, or if the specified user ID is identical to the current one, the list contains all job variables owned by the current user ID that fit into the path name pattern.

Note:

If a catalog ID is specified, it must be embedded between colons : : .
If a user ID is specified, it must be preceded by the dollar sign \$.

Example: LIST JV (1)

1. **LIST JV ASM***

lists all job variables with prefix ASM and owned by the current user ID.

2. **LIST JV \$DEMO.*ASM***

lists all public job variables owned by the user ID DEMO and containing the string ASM in their names.

3. **LIST JV :K:\$DEMO.***

lists all public job variables owned by the user ID DEMO found in catalog **K**.

4. **LIST JV \$DEMO.* NODE=38**

lists all public job variables owned by the user ID DEMO, in a multi-node environment.

An example of a list of BS2000/OSD job variables follows.

Example: LIST JV (2)

The following figure shows an example of a list of BS2000/OSD files generated using the command:

```
LIST JV $DC1.*
```

```

LIST-JV:$DC1.* ----- Row 0 of 8 - Columns 056 064
  COMMAND===>                                SCROLL===> CSR
  PATH-NAME OF JOB VARIABLE
** ***** top of list *****
  :D:$DC1.ASF.TEST.RD-PW
  :D:$DC1.ASF.TEST.WR-PW
  :D:$DC1.INST.JV.NPR
  :D:$DC1.JV.ASF.NINU
  :D:$DC1.JV.SUBMIT.TEST
  :D:$DC1.JV.TEST.NEU
  :D:$DC1.MRS.TEST2
  :D:$DC1.TEMP
** ***** bottom of list *****
    
```

The list appears in Natural ISPF Editor format in BROWSE mode. This means you can use all available BROWSE commands (UP, DOWN, BOTTOM, TOP, LEFT, RIGHT, FIND, LOCATE), and in addition the command SORT.

The list is static: it is filled once and is not updated automatically, for example, after a new job variable has been created.

Lists of job variables can also be created as a result of a CC command, showing the monitoring job variables ('condition codes') of a specific BS2000/OSD job (see the subsection Condition Codes of BS2000/OSD Jobs); in this case, the list also contains the first characters of the values stored in the job variables listed.

Meaning of column headings:

Column	Meaning
PATH-NAME	Full path name including catalog ID and user ID specification
CONTENT	First 64 bytes of job variable value (CC-type lists only)

Line Commands

Select a job variable from a list by typing in a line command in the input field preceding the path name and pressing Enter. Each line command is an abbreviation of a function command. Line commands can also be used as valid abbreviations of function commands entered in the command line.

Available line commands are:

Line Command	Function	Meaning
CH	CHANGE	Change attributes of job variable, for example, define a password.
D	DELETE	Erase job variable from system.
E	EDIT	Start editing job variable, in order to modify its contents or to use cross-copy operations.
I	INFORM	Show attributes and full contents of job variable.
R	RENAME	Rename the job variable.

Local Commands

In Edit Mode:

If you display a job variable in Editor format in EDIT mode, the following local command is available in addition to Editor commands:

Command	Meaning
PASSWORD <password>	If the job variable is password-protected, use this command to enter the valid password in order to update the variable. If you enter the PASSWORD command without parameter, a window prompts you for the password. Password input in the window is invisible.

In List Mode:

If you display lists of job variables in Editor format, you can issue the following local commands in addition to Editor scroll commands: ALL, LAYOUT, RELIST and SORT. For detailed information, see the corresponding subsections of the section Useful Features.

Information about a BS2000/OSD Job Variable

An information screen similar to the following appears as a result of the command:

```
INFORMATION JV $DC1.TEMP
```

```

----- JOB VARIABLE INFORMATION -----
COMMAND ==>>

JOB VARIABLE NAME: $DC1.TEMP

HISTORY                                SECURITY
DATE OF CREATION / UPDATE : 1994-03-02  SHARE                : YES
DATE OF EXPIRATION       : 1994-03-02  ACCESS                 : WRITE
CURRENT LENGTH IN BYTES  : 100          READ-PASSWORD         : NONE
                                WRITE-PASSWORD        : YES

COLUMNS  CURRENT CONTENTS
-----+-----#-----+-----#-----+-----#-----+-----#-----
001-064: P.ADALOD, FNR=140, LIST=L.ADALOD.NOM140, NAME=NOM133-FUSER, INPUT=NO
065-128: M133.FUSER, DEV=T9G, VOL=NOM133, FSEQ=3

```

This screen displays current attributes of job variable \$DC1.TEMP. Its current value is also displayed, if you are authorized to read it; otherwise the subsection titled CURRENT CONTENTS is omitted without notice. The data displayed are self-explanatory.

Local Commands for Job Variable Information

There are two local commands available from this screen:

Local Command	Meaning
PASSWORD <password>	Supplies the correct password for read access, in order to display the subsection CURRENT CONTENTS. If you enter the PASSWORD command without parameter, a window prompts you for the password. Password input in the window is invisible.
REFRESH	Reevaluates data to display latest attributes.

Editing a Job Variable

When a job variable is edited, its contents are read into a single line of an edit session. A screen appears similar to the following:

```

EDIT-JV:$DC1.TEMP ----- Columns 001 072
COMMAND===>                                SCROLL===> CSR
=cols ----+----1----+----2----+----3----+----4----+----5----+----6----+----7--
***** ***** top of data *****
000001 ADALOD,FNR=140,LIST=L.ADALOD.NOM140,NAME=NOM133-FUSER,INPUT=NOM133.FUS
***** ***** bottom of data *****

```

Use the Editor commands RIGHT and LEFT (usually assigned to PF11 and PF10 respectively) to scroll horizontally (the length of the line is restricted to 256 characters).

You can create more than one line of edited data temporarily (this can be useful during complex editing operations), but the SAVE command is rejected if the session contains more than one line of data.

Changing the Attributes of a Job Variable

The CHANGE function command enables you to modify attributes of a job variable. You can specify one or more new attribute values as function parameters, using keyword parameters (see Example 3).

If you issue the CHANGE command with or without object parameters, but without any function parameters or if you issue the line command CH from a list of job variables, a window opens that displays the current attribute values, except for passwords. To modify one or more attributes, overwrite the values in the input fields and press Enter.

Example 1:

The following window opens if you enter the command:

```
CHANGE JV MY.STANDARD.JVAR
```

```

----- BS2000 JOB VARIABLES - ENTRY PANEL -----
COMMAND ==> CHANGE JV MY.STANDARD.JVAR

Job variable ==> *

Password ==> ( If password protected )
Nod +-----CHANGE-JOBVAR: $NATISPF.MY.STANDARD.JVAR-----+
!
! Modify attributes as desired:
!
! SHARED JOB-VARIABLE : YES (YES/NO)
! ACCESS-TYPE : WRITE (READ/WRITE)
! RETENTION-PERIOD : _____ (in days)
! READ-PASSWORD :
! WRITE-PASSWORD :
!
+-----+

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
Help Split End Suspe Rfind Rchan Up Down Swap Left Right Curso
    
```

Meaning of the input fields:

Field	Keyword	Meaning
SHARED JOB-VARIABLE	SHARE	Enter Y or YES to make the job variable accessible to other users. Enter N or NO to restrict access to the owner user ID.
ACCESS-TYPE	ACCESS	Enter W or WRITE to allow write access to the specified job variable, which also implicitly includes authorization for read access. Enter R or READ to allow read access only.
RETENTION-PERIOD	RETPD	Enter a number representing a retention period (in days) that is to be assigned to the specified job variable: This means it can neither be modified nor deleted within the specified period of time.
READ-PASSWORD	READPSWD	Enter a password to protect the job variable against unauthorized read accesses. Enter *NONE to reset an existing password, provided that you specified that password together with the current CHANGE function.
WRITE-PASSWORD	WRITEPSWD	Enter a password to protect the job variable against unauthorized modification or deletion. Enter *NONE to reset an existing password, provided that you specified that password together with the current function.

Example 2:

If the job variable to be changed is password-protected, you must also correctly specify the password. For example, the following window opens if you issue the command:

```
CHANGE JV JV.MIDI.2PTG PASSW=BOBO
```

```

----- BS2000 JOB VARIABLES - ENTRY PANEL -----
COMMAND ==> CHANGE JV JV.MIDI.2PTG PASSW=BOBO

Job variable ==> *

Password      ==>                               ( If password protected )
Nod +-----CHANGE-JOBVAR: $NATISPF.JV.MIDI.2PTG-----+
!
! Modify attributes as desired:
!
! SHARED JOB-VARIABLE : YES           (YES/NO)
! ACCESS-TYPE         : WRITE         (READ/WRITE)
! RETENTION-PERIOD    : _____   (in days)
! READ-PASSWORD       :
! WRITE-PASSWORD      :               (Enter *NONE to reset)
!
+-----+

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Split End  Suspe Rfind Rchan Up    Down Swap Left  Right Curso

```

Enter *NONE to reset the password.

To avoid visible input of the old password on your screen, issue the **CHANGE** command from the BS2000/OSD Files Entry Panel and specify the password in the corresponding input field of the **CHANGE** window.

Example 3:

To avoid being prompted for the new attribute values (for example, during execution of a command script), specify them as function parameters using the keywords indicated above. For example, the command:

```
CH JV PARMJV, SHARE=Y WRITEPSWD=BOHR
```

makes the job variable **PARMJV** shared, that is accessible by other users, and protects it against unauthorized modification.