

Moving Files Between Unix Directories Using External OS Commands



Applies To:

ABAP/4

Summary

Moving large files between UNIX directories using ABAP/4 commands takes a considerable amount of program execution time. You can reduce the time it takes by directly executing external OS commands (Unix Commands) from ABAP/4 code.

The purpose of the sample code contained in this document is to move all the files from the specified source directory in UNIX to the specified destination directory in UNIX using external OS commands. The sample code can be modified to execute any other UNIX command or shell script.

By: Anand Torgal

Company and Title: Wipro Technologies

Date: 25 Feb 2005

Table of Contents

| | |
|--|---|
| Applies To:..... | 1 |
| Summary | 1 |
| Table of Contents | 1 |
| Prerequisite for Executing OS Commands | 2 |
| Step 1 | 2 |
| Step 2 | 2 |
| Program Code | 3 |
| Disclaimer & Liability Notice | 7 |

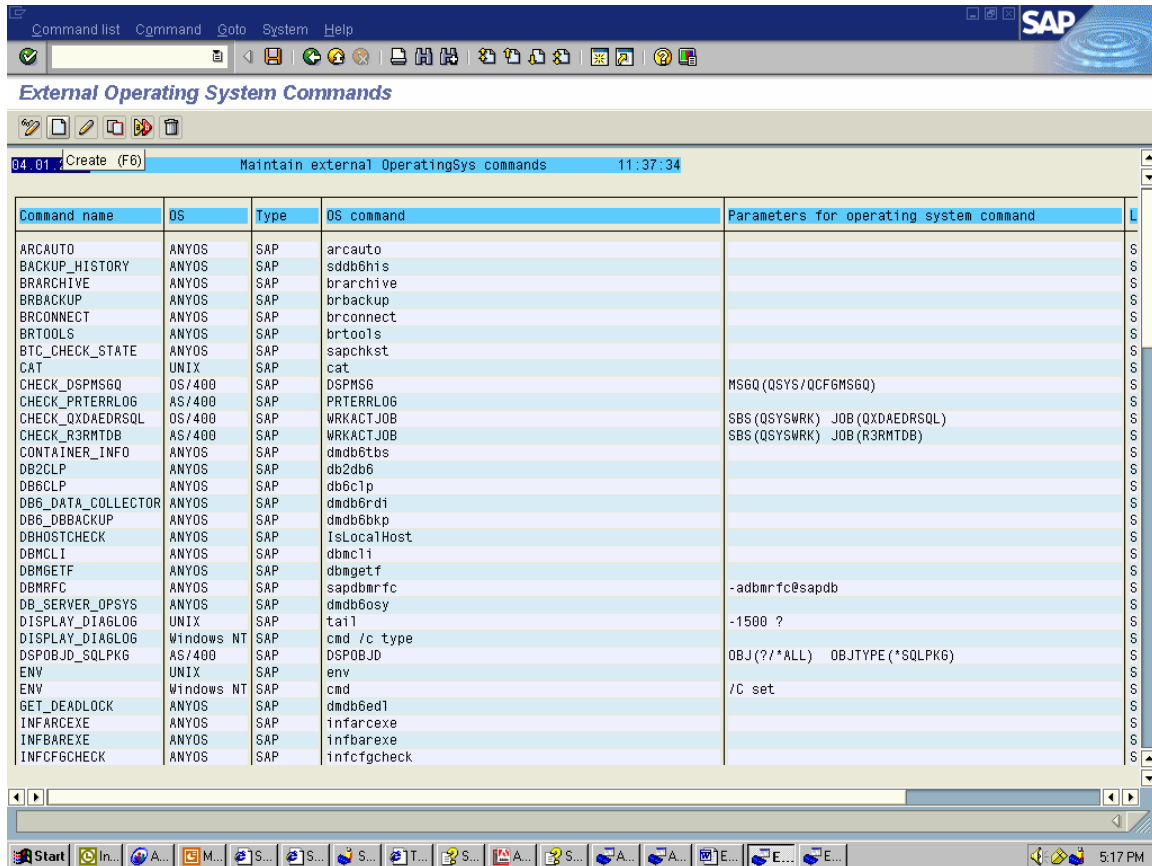
Moving Files Between Unix Directories Using External OS Commands

Prerequisite for Executing OS Commands

Maintain external OS command to be executed.

Step 1

Execute transaction SM69 and press the Create button on the application toolbar in change mode. The SAP user should have appropriate authorizations to execute OS commands.



Step 2

Enter "command name" as "ZMV" (any relevant command name that will be used further in the ABAP program) as shown below. Also specify "Operating system command" as "mv" (UNIX command to be executed from ABAP) and press Save. The UNIX syntax for moving files between two directories is "mv SourceFilePath DestinationFilePath." The source and destination file paths can either be specified in the "Parameters for operating system command" or can be passed from ABAP program dynamically. In the example given, source and destination file paths are passed from ABAP program.

Moving Files Between Unix Directories Using External OS Commands

Program Code

```
REPORT z_move_files_between_unix_dir NO STANDARD PAGE HEADING
MESSAGE-ID ZTEST.
```

```
*-----*
* Program Description:
*
* This program moves all files from source UNIX directory to target
* UNIX directory.
*
* Developer           : Anand Torgal
*-----*
```

```
* SELECTION SCREEN-----*
SELECTION-SCREEN: BEGIN OF BLOCK sel_blk WITH FRAME TITLE text-001.
```

```
PARAMETERS      : p_srcdir LIKE temfile-dirname LOWER CASE OBLIGATORY,
                  "Source Directory
                  p_tgtldr LIKE temfile-dirname  LOWER CASE OBLIGATORY.
                  "Destination Directory
```

```
SELECTION-SCREEN: END OF BLOCK sel_blk.
```

```
*****
*                               Data Declarations
*****
```

Moving Files Between Unix Directories Using External OS Commands

Internal Tables-----

*Internal table for File names

```
DATA BEGIN OF t_xy_filetab OCCURS 0.  
    INCLUDE STRUCTURE emg_fileinfo.  
DATA END OF t_xy_filetab.  
  
DATA t_move LIKE btcxpm OCCURS 0 WITH HEADER LINE.
```

Variables-----

```
DATA: ws_file_path(256) TYPE c, "file path  
      ws_lines TYPE i,  
      ws_zstat LIKE extcmdexex-status, "status returned by  
                                   "sxp_g_command_execute  
      ws_zexit LIKE extcmdexex-exitcode, "exit code returned by  
                                   "sxp_g_command_execue  
      ws_comm_unix LIKE sxpgcolist-parameters, "unix command to  
                                   "sxp_g_command_execute  
      ws_directory LIKE btch0000-text80,  
      ws_length TYPE i.
```

AT SELECTION SCREEN-----

AT SELECTION-SCREEN ON p_srcdir.

* Validate source directory

PERFORM validate_directory USING p_srcdir.

AT SELECTION-SCREEN ON p_tgtdir.

* Validate target directory

PERFORM validate_directory USING p_tgtdir.

* START OF SELECTION-----*

START-OF-SELECTION.

REFRESH t_xy_filetab[].

* Get all the file names from source directory

CALL FUNCTION 'ISU_M_GET_FILES_OF_DIR'
EXPORTING

 x_path = p_srcdir

TABLES

 xy_filetab = t_xy_filetab

EXCEPTIONS

 no_start = 1

 no_next = 2

 no_finish = 3

 general_error = 4

 OTHERS = 5.

IF sy-subrc <> 0.

 MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno
 WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.

 EXIT.

ENDIF.

CLEAR ws_lines.

DESCRIBE TABLE t_xy_filetab LINES ws_lines.

IF ws_lines <= 0.

 MESSAGE i888 WITH 'No files found in directory' p_srcdir.

 EXIT.

ENDIF.

Moving Files Between Unix Directories Using External OS Commands

```
* Place '/' at the end of source directory if not present
CLEAR ws_length.
ws_length = strlen( p_srcdir ).
ws_length = ws_length - 1.
IF p_srcdir+ws_length NE '/'.
  CONCATENATE p_srcdir '/' INTO p_srcdir.
ENDIF.

* Move all the files to target directory
LOOP AT t_xy_filetab.

  CLEAR: ws_file_path,
        ws_comm_unix,
        ws_zstat,
        ws_zexit.
  REFRESH t_move[].

* Construct source file path
  CONCATENATE p_srcdir t_xy_filetab-name
    INTO ws_file_path.

* Forming the command as per unix syntax to perform move in unix
  CONCATENATE ws_file_path p_tgt_dir
    INTO ws_comm_unix SEPARATED BY space.

* Call function to execute 'ZMV' command defined in SM69
  CALL FUNCTION 'SXPG_COMMAND_EXECUTE'
    EXPORTING
      commandname                = 'ZMV'
      additional_parameters       = ws_comm_unix
      operatingsystem             = sy-opsys
      stdout                      = 'X'
      stderr                     = 'X'
      terminationwait            = 'X'
    IMPORTING
      status                     = ws_zstat
      exitcode                   = ws_zexit
    TABLES
      exec_protocol              = t_move
    EXCEPTIONS
      no_permission              = 1
      command_not_found          = 2
      parameters_too_long        = 3
      security_risk              = 4
      wrong_check_call_interface = 5
      program_start_error        = 6
      program_termination_error  = 7
      x_error                    = 8
      parameter_expected         = 9
      too_many_parameters        = 10
      illegal_command            = 11
      wrong_asynchronous_parameters = 12
      cant_enq_tbtco_entry       = 13
      jobcount_generation_error  = 14
      OTHERS                     = 15.
  IF sy-subrc <> 0 OR ws_zexit <> 0.
    MESSAGE ID sy-msgid TYPE 'I' NUMBER sy-msgno
      WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.
  ENDIF.
```

Moving Files Between Unix Directories Using External OS Commands

```
ENDLOOP.
```

```
END-OF-SELECTION.
```

```
FREE:  t_move[],  
       t_xy_filetab[].
```

```
*&-----*  
*&      Form  VALIDATE_DIRECTORY  
*&-----*  
*      Validate input directory  
*-----*  
*      -->P_DIR  text  
*-----*
```

```
FORM validate_directory USING    p_dir.
```

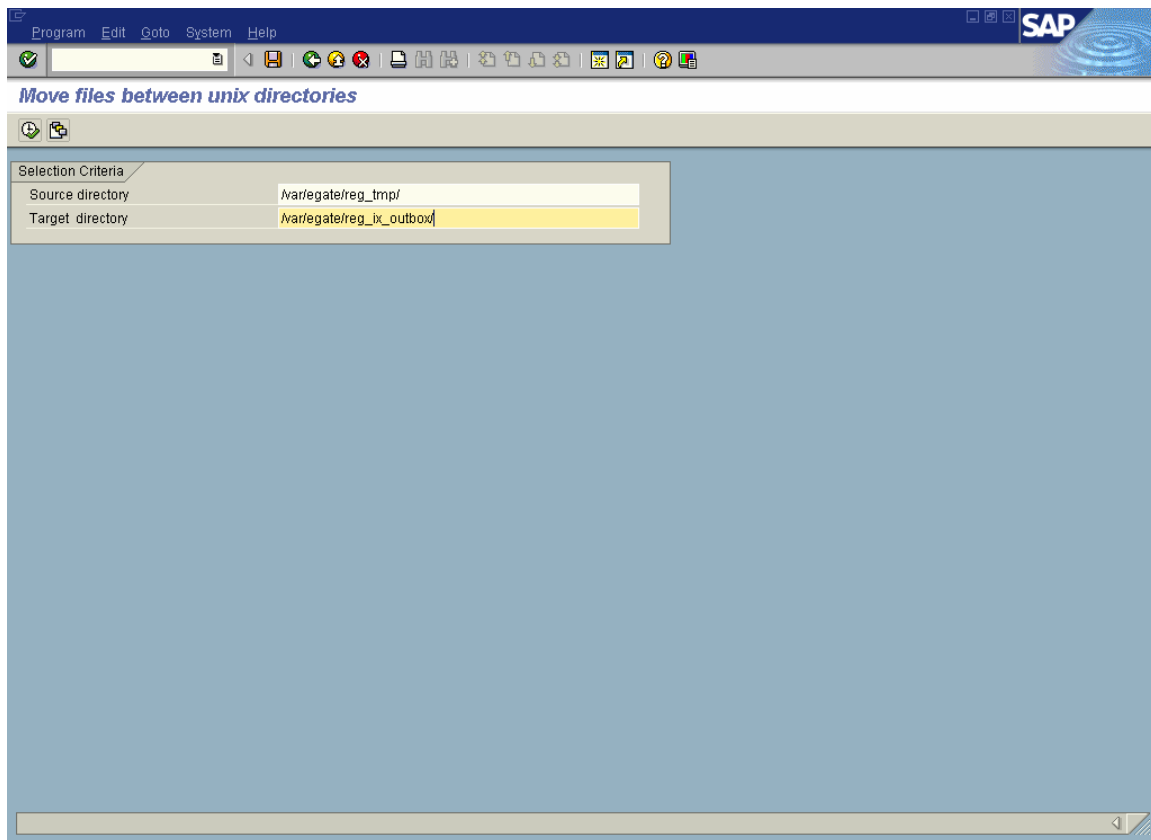
```
ws_directory = p_dir.
```

```
CALL FUNCTION 'PFL_CHECK_DIRECTORY'  
  EXPORTING  
    directory                = ws_directory  
  EXCEPTIONS  
    pfl_dir_not_exist       = 1  
    pfl_permission_denied   = 2  
    pfl_cant_build_dataset_name = 3  
    pfl_file_not_exist      = 4  
    OTHERS                  = 5.
```

```
IF sy-subrc <> 0.  
  MESSAGE e888 WITH 'Invalid directory' ws_directory.  
ENDIF.
```

Enter source and destination directories on the selection screen and execute. The files moved from source directory to destination directory can be viewed using transaction code AL11.

Moving Files Between Unix Directories Using External OS Commands



Disclaimer & Liability Notice

This document may discuss sample coding, which does not include official interfaces and therefore is not supported. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing of the code and methods suggested here, and anyone using these methods, is doing it under his/her own responsibility.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of the technical article, including any liability resulting from incompatibility between the content of the technical article and the materials and services offered by SAP. You agree that you will not hold SAP responsible or liable with respect to the content of the Technical Article or seek to do so.

Copyright © 2005 SAP AG, Inc. All Rights Reserved. SAP, mySAP, mySAP.com, xApps, xApp, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product, service names, trademarks and registered trademarks mentioned are the trademarks of their respective owners.