CERES Software Bulletin 95-16

Runtime Parameter and I/O File Name Retrieval from PCF

Alice Fan (t.f.fan@larc.nasa.gov)

1.0 Purpose

This bulletin describes the Fortran 90 wrapper and the UNIX command line functions to retrieve runtime parameters and file names specified in the Process Control File (PCF) from a Fortran program and the UNIX script levels. The Science Data Production (SDP) Toolkit Process Control Tools perform the task of communicating PCF information to the Product Generation Executive (PGE). This information may consist of Production Run ID, Science Software ID, metadata and attributes, runtime parameters, and physical file names.

2.0 What is a Process Control File?

The PCF is the central source of information to the Process Control (PC) tools. Each PGE requires one PCF and this file is to be pointed to by the PGS_PC_INFO_FILE environment variable in the run script. Every entry in the PCF starts with a number called the logical identifier. Each logical identifier corresponds to one or more physical references. The software accesses this information by the logical identifier. Appendix A explains the PCF format. Appendix B is a simplified cloud PCF.

3.0 What is a Process Control Tool?

The Process Control Tool is the second level of SDP Toolkit (Status Message File tool is the lowest level). There are two sets of PC tools: the command tools, which are callable from the UNIX shell scripts, and the Application Program Interface tools, callable from Fortran, C, or Ada programs. Much of the functionality is duplicated in these two groups. Section 4 explains the f90 wrapper which is callable from the source code and Section 5 explains the command level tools for retrieving runtime parameters and file names from the script. Both sections use the PCF shown in Appendix B as an input file.

4.0 PCF f90 Wrapper Routine

The PCF module makes the following PUBLIC routines available to the user:

    (1) GetParam   (LogicID, Answer, status)

    (2) GetFileName   (LogicID, FileName, status, version)

    (3) GetNumOfFile (LogicID, NumOfFile, status)

The GetParam routine takes the input parameter, "LogicID", and outputs the value associated with this logical identifier to the parameter "Answer" in the format of one byte, two byte, or four byte INTEGER, or four byte or eight byte REAL, or STRING, depending on the data type speci-

fied for the "Answer" parameter in the calling statement. A status of OK or not OK is also returned. OK is defined in the CERESLIB ceres_status module with a value of 0.

The GetFileName routine takes the input "LogicID" and optional version (or sequence) number and returns the full path and file name to the parameter "FileName." The default sequence number is set to 1. A status of OK or not OK is also returned.

The GetNumOfFiles routine takes the input "LogicID" and returns the number of files to the integer parameter "NumOfFile" and a status of OK or not OK. Section 4.1 shows calling examples for the above three wrappers in a test program. Section 4.2 is the output for the test program.

4.1 Calling Examples in the Test Program

```
PROGRAM test_pcf
!CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
! A driver to test the PCF toolkit interface
!CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
USE  PCF, ONLY : GetParam, GetFileName, GetNumOfFiles
USE  MSG, ONLY :WriteStatus
USE  f90_kind, ONLY :  real4, real8
USE  CERES_STATUS, ONLY : OK
IMPLICIT NONE
INCLUDE 'PGS_CERES_25000.f'
!
!    Users need to use the logical numbers in their own PC file.
!    For this test program to work, the following lines must exist in the  PC file
!    101|AVHRR_861001_03|/CERES/data/PGE4.1/Input/Gac||||3
!    101|AVHRR_861001_04|/CERES/data/PGE4.1/Input/Gac||||2
!    101|AVHRR_861001_05|/CERES/data/PGE4.1/Input/Gac||||1
!    601|DateOfData|861001
!    604|Num Chunks to Process|20
!    615|Solar Zenith Angle Day/Night Threshold|85.0
!    616|Latitude Polar/Non-Polar Threshold|60.0

INTEGER, PARAMETER :: CIDinput=101
INTEGER, PARAMETER :: DataDate=601
INTEGER, PARAMETER :: NUMCHUNK=604
INTEGER, PARAMETER :: MAXSOLARZENITH=615
INTEGER, PARAMETER :: MAXPOLARLATITUDE=616

INTEGER          :: status, ianswer, idumb, filenum, i
REAL(real4)      :: r4answer
REAL(real8)      :: r8answer
```

```fortran
      CHARACTER (len=200):: sanswer, filenm

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! get an INTEGER run-time parameter from PCF
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      call GetParam (NUMCHUNK, ianswer, status)

        if (status .ne. OK ) then
          call WriteStatus(CERES_E_GETINTPARAM_UNKNOWN,   &
             StrVar = " - for logical id = %i5", IVar1=NUMCHUNK)
        else
           print*,"Number of chunk = ",ianswer
         end if
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! get a  REAL  8 run-time parameter from PCF
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      call GetParam(MAXPOLARLATITUDE, r8answer, status)

        if (status .ne. OK) then
          call WriteStatus(CERES_E_GETREALPARAM_UNKNOWN,   &
             StrVar = " - for logical id = %i5", IVar1=MAXPOLARLATITUDE)
        else
          print*,"Maxi Polar Latitude = ",r8answer
        end if
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! get a  REAL 4 run-time parameter from PCF
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      call GetParam(MAXSOLARZENITH, r4answer, status)

        if (status .ne. OK) then
          call WriteStatus(CERES_E_GETREALPARAM_UNKNOWN,   &
             StrVar = " - for logical id = %i5", IVar1=MAXSOLARZENITH)
        else
          print*,"Maxi Solar Zenith = ",r4answer
        end if
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! get a STRING run-time parameter from PCF
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
      call GetParam(DataDate, sanswer, status)

        if (status .ne. OK) then
```

```fortran
            call WriteStatus(CERES_E_GETSTRPARAM_UNKNOWN,   &
                StrVar = " - for logical id = %i5", IVar1=DataDate)
        else
            print*,"Data Date as string = ",sanswer
        end if
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! get Number of files from PCF
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
call Getnumoffiles(CIDinput, filenum, status)

    if (status .ne. OK) then
        call WriteStatus(CERES_E_GETNUMOFFILE_UNKNOWN,   &
            StrVar = " - for logical id = %i5", IVar1=CIDinput)
    else
        print*,"Number of file is : ",filenum
    end if
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! get FILE NAME from PCF
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 do i = 1, filenum
     call GetFileName(CIDinput, filenm, status, i)

     if (status .ne. OK) then
     call WriteStatus(CERES_E_GETFILENAME_UNKNOWN,   &
         StrVar = " - for logical id = %i5", IVar1=CIDinput)
     else
         print*,"file name is : ", filenm
     end if
 end do
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! get an INTEGER run-time parameter from PCF - incorrect logical number
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
idumb = 990
call GetParam(idumb, ianswer, status)

    if (status .ne. OK) then
        call WriteStatus(CERES_E_GETINTPARAM_UNKNOWN,   &
            StrVar = " - for logical id = %i5", IVar1=990)
    else
        print*,"Data Date = ",ianswer
    end if
```

END PROGRAM test_pcf

4.2 Outputs from the Test Program

Number of chunk = 1

Maxi Polar Latitude =   60.0000000000000000

Maxi Solar Zenith =   85.0000000

Data Date as string = 861001

Number of file is : 3

filename is:/home/saisun15-1h/fan/CLOUD8/data/Gac/AVHRR_861001_03

filename is:/home/saisun15-1h/fan/CLOUD8/data/Gac/AVHRR_861001_04

filename is:/home/saisun15-1h/fan/CLOUD8/data/Gac/AVHRR_861001_05

5.0 UNIX Script level Routines

The corresponding routines to get the runtime parameters and file names from a script are:

(1) PGS_PC_GetConfigDataCom   -   get run time parameter value for a logical ID

(2) PGS_PC_GetReferenceCom   -   get the file name for a logical ID

(3) PGS_PC_GetNumberOfFilesCom   -   get number of files for a logical ID

Section 5.1 shows three calling examples and Section 5.2 are the outputs from this test script. The "Set 1" command lines get the runtime parameter "Spatial Coherence Tile Size" into an environment variable "SPsize." For this to work, it is assumed that the line: **630|Horizontal tile factor for spatial Coherence|64** exists in the PCF.

The "Set 2" command lines get the file name for logical ID 101 into an environment variable "filename." For this to work, it is assumed that the line: **101|AVHRR_861001_03|/CERES/data/ PGE4.1/Input/Gac||||3** exists in the PCF file.

The "Set 3" command lines get number of files into an environment variable "NumOfFile." For this to work, it is assumed that the following three lines exist in the PC file:

101|AVHRR_861001_03|/CERES/data/PGE4.1/Input/Gac||||3

101|AVHRR_861001_04|/CERES/data/PGE4.1/Input/Gac||||2

101|AVHRR_861001_05|/CERES/data/PGE4.1/Input/Gac||||1

5.1 Calling Examples for the Above Three Routines

```
#!/bin/csh
setenv MYHOME $HOME
set my_path=($path)
setenv PGSDIR /opt/net/PGSTKV5
```

```
source $PGSDIR/bin/pgs-dev-env.csh
set path= ($my_path $pgs_path)
setenv PGS_PC_INFO_FILE $MYHOME/f90_cld/PCfile
setenv PGSMSG /opt/net/cereslib/Message
#      Set 1
set logicalID = 630
setenv SPsize `$PGSDIR/bin/PGS_PC_GetConfigDataCom $logicalID`
echo "Spatial Coherence Tile Size = $SPsize"
#     Set 2
set LID = 101
set ver = 1
set xx = `$PGSDIR/bin/PGS_PC_GetReferenceCom $LID $ver`
set filename = $xx[1]
set version  = $xx[2]
echo File Name = $filename
echo Next Version number = $version
#      Set 3
setenv NumOfFile `$PGSDIR/bin/PGS_PC_GetNumberOfFilesCom 101`
echo "Number of files for 101 =  $NumOfFile"
#
# check if file exist ?
if (( ! -e $filename) || ( -z $filename )) then
   echo " $filename DOES NOT EXIST"
   exit (1)
endif
```

5.2 Outputs from the Test Script

Set 1:

Spatial Coherence Tile Size = 64

Set 2:

File Name = /home/saisun15-1h/fan/CLOUD8/data/Gac/AVHRR_861001_03

Next Version number = 2

Set 3 :

Number of files for 101 = 3

Appendix A: Constructing a PC file

A template PCF file, PCF.v5, resides under directory $PGSDIR/runtime is available for reference. ($PGSDIR is /opt/net/PGSTK on thunder computer and /opt/net/pGSTKV5 on saisun00 computer). There are ten sections in a PC file. Each section starts with "?" in the first column and subject name. These subject names must not be changed nor deleted.

    ?   SYSTEM RUNTIME PARAMETERS
    ?   PRODUCT INPUT FILES
    ?   PRODUCT OUTPUT FILES
    ?   SUPPORT INPUT FILES
    ?   SUPPORT OUTPUT FILES
    ?   USER DEFINED RUNTIME PARAMETERS
    ?   INTERMEDIATE INPUT
    ?   INTERMEDIATE OUTPUT
    ?   TEMPORARY IO
    ?   END

The SYSTEM RUNTIME PARAMETERS section contains only two entries. Each can have up to 200 characters. They are:

    # --
    # Production Run ID - unique production instance identifier
    # --
    1
    # --
    # Software ID - unique software configuration identifier
    # --
    1

All INPUT/OUTPUT sections contains entries with seven fields and look like:

    240|Rtmp_861001_03|/CERES/data/PGE4.1/Input/MOA||||1

    field 1: logic identifier (240)

    field 2: file name (Rtmp_861001_03)

    field 3: directory (/CERES/data/PGE4.1/Input/MOA)

    fields 4 and 5: two reserved fields (blank)

    field 6: attribute location (the name of a file that contains data about the file in field 2)

    field 7: sequence number

The logical identifier numbers 9992, 9993, 9994, 9995, 9996 are reserved by CERES library and the numbers between 10,000 to 10,999 are reserved exclusively for Toolkit use. The entire length of any entry may not exceed 500 characters.

Each of the INPUT/OUTPUT section can specify one default directory starting with "!". This may be overwritten for individual files by inserting the fully-qualified directory. For example, the

---

following first four lines specify that all the three AVHRR files are under directory "/CERES/data/PGE4.1/Input/Gac." The fifth line specifies the file "Rtmp_861001_03" is under directory "CERES/data/PGE4.1/Input/MOA."

> **! /CERES/data/PGE4.1/Input/Gac**
>
> **101|AVHRR_861001_03|||||3**
>
> **101|AVHRR_861001_04|||||2**
>
> **101|AVHRR_861001_05|||||1**
>
> **240|Rtmp_861001_03|/CERES/data/PGE4.1/Input/MOA||||1**

The USER DEFINED RUNTIME PARAMETERS consists of entries with three fields and look like:

> **630|Horizontal tile factor for spatial Coherence|64**

field 1: logical identifier

field 2: parameter name, an optional text string of up to 200 characters.

filed 3: parameter value, a string of up to 200 characters

The END section has no entry at current stage.

Appendix C shows the steps of using a utility called "ceresutil" in the CERESLIB directory (/opt/net/cereslib) to check the correctness of a PCF file. The ceresutil utility is a menu driven shell script to initiate two Toolkit utilities: pccheck and smfcompile. Please see bulletin 96-14 for the usage of smfcompile utility. To use ceresutil, just copy it from /opt/net/cereslib and type "ceresutil" at the UNIX prompt, or type "/opt/net/cereslib/ceresutil" to initiate.

Appendix B: A Simplified Cloud Process Control File

Environment variable PGS_PC_INFO_FILE must point to this file

#

? SYSTEM RUNTIME PARAMETERS

Production Run ID - unique production instance identifier

1

Software ID - unique software configuration identifier

1

#

? PRODUCT INPUT FILES

The next line specifies the default directory for all the input files.

The default directory is overwritten when the directory is specified

---

! /CERES/data/PGE4.1/Input/Gac

#

101|AVHRR_861001_03|||||3
101|AVHRR_861001_04|||||2
101|AVHRR_861001_05|||||1

The effect of the next three lines is the same as the above four lines combined

---

###101|AVHRR_861001_03|/CERES/data/PGE4.1/Input/Gac||||3

###101|AVHRR_861001_04|/CERES/data/PGE4.1/Input/Gac||||2

###101|AVHRR_861001_05|/CERES/data/PGE4.1/Input/Gac||||1

The next two lines specify their own path.

240|Rtmp_861001_03|/CERES/data/PGE4.1/Input/MOA||||1

241|Pres_861001_03|/CERES/data/PGE4.1/Input//MOA||||1

#

? PRODUCT OUTPUT FILES

#

! /home/saisun15-1h/fan/CLOUD8/data

Cloud main output file

```
# ---------------------------------------------------------------
302|Cookie_861001_03|/home/saisun15-1h/fan/CLOUD8/data/Cookie||||
303|CloudVal_861001_03|/home/saisun15-1h/fan/CLOUD8/data/CloudVal||||
304|Msg_861001_03_Error|/home/saisun15-1h/fan/CLOUD8/data/Messages||||
#
?  SUPPORT INPUT FILES
#
! /opt/net/PGSTK5/runtime
#
?  SUPPORT OUTPUT FILES
# ---------------------------------------------------------------
# These files support the SMF log functionality. Each run will cause status information to be written to
# 1 or more of the Log files. To simulate DAAC operations, remove the 3 Log files between test runs.
# Remember: all executables within a PGE will contribute status data to the same batch of log files.
# ---------------------------------------------------------------
! /home/saisun15-1h/fan/f90_log
10100|LogStatus|/home/saisun15-1h/fan/f90_log||||1
10101|LogReport|/home/saisun15-1h/fan/f90_log||||1
10102|LogUser|/home/saisun15-1h/fan/f90_log||||1
10103|TmpStatus|/home/saisun15-1h/fan/f90_log||||1
10104|TmpReport|/home/saisun15-1h/fan/f90_log||||1
10105|TmpUser|/home/saisun15-1h/fan/f90_log||||1
10110|MailFile|/home/saisun15-1h/fan/f90_log||||1
# ---------------------------------------------------------------
?  USER DEFINED RUNTIME PARAMETERS
# ---------------------------------------------------------------
509|Latitude threshold for Polar/non_Polar region|85.0
#
601|DateOfData|861001
602|HourOfData|3
603|SatelliteCode|9
#
604|Num Chunks to Process|20
605|Scanlines per chunk|64
606|Initial Scanlines to skip|2
607|Pixels per scanline|409
608|Data type : AVHRR=1,  VIRS=2, MODIS=3|1
#
615|Solar Zenith Angle Day/Night Threshold|85.0
616|Latitude Polar/Non-Polar Threshold|60.0
#
```

630|Horizontal tile factor for spatial Coherence|64

631|Vertical   tile factor for spatial Coherence|64

632|Horizontal tile factor for classification|8

633|Vertical   tile factor for classification|8

701|Execute SERCAA Tests|n

702|Execute Welch|n

706|Execute Stowe|n

#

801|Generate Cookie Dough Data Product|y

9992|Satellite|TRMM

9993|Instrument|CERES

9994|System Release|1

9995|Software Version|Version 9

9996|Product name|Cookie Dough

These parameters are required to support the PGS_SMF_Send* tools. If the first parameter

(TransmitFlag) is disabled, then none of the other parameters need to be set. By default, this

functionality has been disabled. To enable, set TransmitFlag to 1 and supply the other 3

parameters with local information.

10109|TransmitFlag; 1=transmit,0=disable|0

10106|RemoteHost|thunder

10107|RemotePath|/disk2/thunder/fan

10108|EmailAddresses|fan@saisun15

#

? INTERMEDIATE INPUT

#

! /home/saisun15-1h/fan/f90_log

#

? INTERMEDIATE OUTPUT

#

! /home/saisun15-1h/fan/f90_log

#

? TEMPORARY IO

#

! /home/saisun15-1h/fan/f90_log

#

#

? END

Appendix C: Using the ceresutil to Check a PCF

fan@saisun15 % /**opt/net/cereslib/ceresutil**

 CERES UTIL

----

 1. Compile message status file
 2. Check PCF file correctness

 3. Create a message File - not ready yet
 Q. Quit

 Please enter option: **2**

 Enter PCF file name (include path, if not under current dir)--> **~/f90_cld/PCfile**

Warning - possible problem in default file location.

Line number: 101

Check of /home/saisun15-1h/fan/f90_cld/PCfile completed

Errors found: 0

Warnings found: 1

 Exiting CERESUTIL, Goodbye !

Note: The warning message means a specified default directory (using !) may be write protected."