Clouds and the Earth's Radiant Energy System (CERES)

Data Management System

CERES Inversion to Instantaneous TOA Fluxes and Empirical Estimates of Surface Radiation Budget Subsystems 4.5 and 4.6

Release 2 Test Plan
Version 2

Primary Authors

Sandy Nolan, Carla Franklin

Science Applications International Corporation
One Enterprise Parkway, Suite 300
Hampton, VA 23666

Data Management Office
Atmospheric Sciences Division
NASA Langley Research Center
Hampton, VA 23681-0001

February 1998

TABLE OF CONTENTS

Section Page

1.0 Introduction 1
1.1 Document Overview 1
1.2 Subsystem Overview 2
1.2.1 CERES Inversion to Instantaneous TOA Fluxes and EmpiricalEstimates of Surface Radiation Budget Subsystems 4.5 and 4.6 Main Processor . . . 2
1.2.2 CERES Inversion to Instantaneous TOA Fluxes and Empirical Estimates of Surface Radiation Budget Subsystems 4.5 and 4.6 Postprocessor 2
2.0 Test Environment 3
2.1 External Interface 3
2.2 Directory Structure and File Descriptions 4
3.0 Software and Data File Installation Procedures 5
3.1 Installation 5
3.2 Compilation 6
4.0 Test and Evaluation Procedures 7
4.1 Stand Alone Test Procedures 7
4.1.1 Main Processor PCF Generator 7
4.1.2 Main Processor Execution 8
4.1.3 Postprocessor PCF Generator 9
4.1.4 Postprocessor Execution 10
4.2 Normal Operating Procedures 10
4.3 Evaluation Procedures 10
4.3.1 Exit Codes 10
4.3.2 Log and Status File Results 11
4.3.3 Metadata Evaluation 11
4.3.4 Execution of Comparison Software 11
4.3.5 Evaluation of Comparison Software Output 12
4.3.6 Evaluation of SSF HDF Product 12
4.4 Solutions to Possible Problems 13
APPENDIX A Acronyms and Abbreviations A-1
APPENDIX B Directory Structure Diagram B-1
APPENDIX C File Description Tables C-1

LIST OF FIGURES

Figure Page

Figure B-1. Directory Structure for the Inversion Tar File B-1

LIST OF TABLES

Table | Page

Table 2-1. CERESlib Routines Used by Inversion 3
Table C.1-1. Production Scripts and Executables C-1
Table C.2-2. Process Control Files C-2
Table C.2-1. Metadata Control Files C-2
Table C.3-1. Production Makefiles C-2
Table C.4-1. Ancillary Input Data C-3
Table C.5-1. Primary Input Data C-3
Table C.6-1. Expected Output Data C-4
Table C.7-1. Production Output Data C-5
Table C.8-1. Output Temporary Data Files C-6
Table C.9-1. Error and Status Message Files C-6
Table C.10-1. Test Evaluation Scripts and Executables C-7
Table C.10-2. Test Evaluation Source Code and Makefile C-7
Table C.10-3. Test Evaluation Output (Expected Results) C-7
Table C.10-4. Test Evaluation Output (Test Results) C-8

1.0 Introduction

The Clouds and the Earth's Radiant Energy System (CERES) is a key component of the Earth Observing System (EOS). The CERES instruments are improved models of the Earth Radiation Budget Experiment (ERBE) scanner instruments, which operated from 1984 through 1990 on the National Aeronautics and Space Administration's (NASA) Earth Radiation Budget Satellite (ERBS) and on the National Oceanic and Atmospheric Administration's (NOAA) operational weather satellites NOAA-9 and NOAA-10. The strategy of flying instruments on Sun-synchronous, polar orbiting satellites, such as NOAA-9 and NOAA-10, simultaneously with instruments on satellites that have precessing orbits in lower inclinations, such as ERBS, was successfully developed in ERBE to reduce time sampling errors. CERES will continue that strategy by flying instruments on the polar orbiting EOS platforms simultaneously with an instrument on the Tropical Rainfall Measuring Mission (TRMM) spacecraft, which has an orbital inclination of 35 degrees. In addition, to reduce the uncertainty in data interpretation, and to improve the consistency between the cloud parameters and the radiation fields, CERES will include cloud imager data and other atmospheric parameters. The first CERES instrument is scheduled to be launched on the TRMM spacecraft in 1997. Additional CERES instruments will fly on the EOS Morning Crossing Mission (EOS-AM) platforms, the first of which is scheduled for launch in 1998, and on the EOS Afternoon Crossing Mission (EOS-PM) platforms, the first of which is scheduled for launch in 2000.

1.1 Document Overview

This document, CERES Inversion to Instantaneous Top-of-Atmosphere (TOA) Fluxes and Empirical Estimates of Surface Radiation Budget Subsystems 4.5 and 4.6 Release 2 Test Plan, is part of the CERES Subsystems 4.5 and 4.6 Release 2 delivery package provided to the Langley Distributed Active Archive Center (DAAC). It provides a description of the CERES Inversion to Instantaneous TOA Fluxes and Empirical Estimates of Surface Radiation Budget Release 2 software and explains the procedures for installing, executing, and testing the software. A section is also included on validating the software results. A description of acronyms and abbreviations is provided in Appendix A, directory structure diagrams are contained in Appendix B and a description of all the software and data files is contained in Appendix C.

This document is organized as follows:

Section 1.0 - Introduction
Section 2.0 - Test Environment
Section 3.0 - Software and Data File Installation Procedures
Section 4.0 - Test and Evaluation Procedures
Appendix A - Acronyms and Abbreviations
Appendix B - Directory Structure Diagrams
Appendix C - File Description Tables

1.2 Subsystem Overview

1.2.1 CERES Inversion to Instantaneous TOA Fluxes and Empirical Estimates of Surface Radiation Budget Subsystems 4.5 and 4.6 Main Processor

CERES Inversion to Instantaneous TOA Fluxes and the Empirical Estimates of Surface Radiation Budget, Subsystems 4.5 and 4.6, converts CERES filtered radiance measurements to instantaneous radiative flux estimates at the top of the Earth's atmosphere and produces radiative flux estimates at the Earth's surface for each CERES footprint.

The Inversion Subsystem 4.5 calculates estimates of the radiant flux at the TOA based on input from the Preliminary Single Satellite Footprint (PRE_SSF) produced by the Convolution of Imager Cloud Properties with CERES Footprint Point Spread Function (PSF) Subsystem 4.4. This inversion process is dependent on several factors, including Earth surface features, the extent of cloudiness, and the relative geometry of the spacecraft, the Sun, and the measurement field-of-view. Each radiometric measurement is spectrally corrected to give an unfiltered measurement. The observed scene is determined by a scene identification algorithm based on land or sea types and cloud cover. Estimates of the radiant flux at the TOA are computed based on scene information, geometrical considerations, and the unfiltered measurements.

The Surface Flux Estimation, Subsystem 4.6, calculates estimates of radiant flux at the Earth's surface based on TOA fluxes calculated in Subsystem 4.5. Additional input data required by the Subsystem include precipitable water, surface emissivity, and cloud properties from the PRE_SSF, and pressure, temperature, and humidity profiles from a Meteorological, Ozone, and Aerosol (MOA) Product. Model A Shortwave (SW) net and downward surface fluxes are estimated using the Li-Leighton algorithm. Model B SW downward surface fluxes are estimated using the Staylor algorithm.. Model A Longwave (LW) net and downward surface fluxes are estimated using the Ramanathan-Inamdar Algorithm, if the scene type is clear-sky over ice-free ocean or clear-sky over land in the tropics. Model B LW net and downward surface fluxes are estimated using the Gupta Algorithm for all scene types.

The output of the Subsystems 4.5 and 4.6 Main Processor consists of a binary SSF product, which serves as input for CERES Subsystem 5.0 and Subsystem 9.0, an ASCII Quality Control (QC) report, and a binary QC file.

1.2.2 CERES Inversion to Instantaneous TOA Fluxes and Empirical Estimates of Surface Radiation Budget Subsystems 4.5 and 4.6 Postprocessor

CERES Inversion to Instantaneous TOA Fluxes and the Empirical Estimates of Surface Radiation Budget, Subsystems 4.5 and 4.6, Postprocessor reads the binary SSF product as input and generates an SSF product in Hierarchical Data Format (HDF).

2.0 Test Environment

2.1 External Interface

The input data provided for Subsystems 4.5 and 4.6 testing contains ERBE data in a simulated CERES format. There will be no changes in external interfaces for Subsystems 4.5 and 4.6, when production data is used in place of the simulated test data.

The CERES library (CERESlib) Fortran 90 modules located in libraries cereslib.a and data_products.a and used by Subsystems 4.5 and 4.6 software are listed in Table 2-1.

Table 2-1. CERESlib Routines Used by Inversion (1 of 2)

File Name	Description
f90_kind	Provides Fortran 90 (F90) compiler-specific KIND values
ceres_status	Provides a common set of file and return statuses
reference_grid	Provides an interface to the CERES reference grid
ceres_constants	Provides commonly used CERES constants
ceres_defaults	Provides system-defined CERES default values
ceres_indices	Provides commonly used array indices
ceres_time	Provides commonly used time conversions
qcheader	Provides utilities for writing QC header
pcf	Provides run-time parameters from a Processing Control File (PCF)
io	Provides Toolkit input/output wrappers
msg	Provides interface to the Toolkit Status Message Files (SMF)
cadm_new	Provides subroutines to read and interpolate CERES SW and LW anisotropic models
ssf_meta	Provides subroutines to read and write metadata to SSF and SSF QC products
surf_typdef	Provides F90 structures to interface with surface flux estimation algorithms
surf_sw_model_a	Provides Li-Leighton SW surface flux algorithm
surf_sw_model_b	Provides Staylor SW surface flux algorithm
surf_lw_model_a	Provides Ramanathan-Inamdar LW surface flux algorithm

Table 2-1. CERESlib Routines Used by Inversion (2 of 2)

File Name	Description
surf_lw_model_b	Provides Gupta LW surface flux algorithm
ssf_typdef	Provides SSF F90 structure definition and subroutines to read, write, open, and close SSF files
ssfqc_typdef	Provides SSF QC F90 structure definition and subroutines to read, write, open, and close SSF QC files
range_check	Provides general range checking functions
moa_io	Provides Meteorological, Ozone, and Aerosol (MOA) F90 structure definition and subroutines to read, write, open, and close MOA files

2.2 Directory Structure and File Descriptions

The CERES Inversion to Instantaneous TOA Fluxes and Empirical Estimates of Surface Radiation Budget Subsystems 4.5 and 4.6 Release 2 delivery package contains four tar files, inversion_test_plan_R2-45.tar.Z, inversion_src_R2-45.tar.Z, inversion_anc_R2-45.tar.Z, and inversion_data_R2-45.tar.Z. The directory structures of the untarred files are shown in Appendix B. The contents of the tar files are categorized according to software files, ancillary data files, documentation files, and test data files. A description of each file included in the delivery package can be found in Tables C.1-1 through C.10-4 in Appendix C.

3.0 Software and Data File Installation Procedures

This section describes how to install the Subsystems 4.5 and 4.6 Inversion software in preparation for making the necessary test runs at the Langley DAAC. The installation procedures include instructions for uncompressing and untarring the delivered tar files, properly defining environmental variables, and recompiling the Inversion programs.

3.1 Installation

Software/Data File Install Procedure:

1. The scripts, makefiles, and Process Control Files in the Subsystems 4.5 and 4.6 delivery package expect the CERES environment variable, **$CERESENV**, to point to a file which sets the following environment variables:

PGSDIR	**- Directory for Toolkit libraries**
F90	**- Pointer to the SGI F90 64 bit compiler**
CERESHOME	**- Top Directory for CERES Software**
CERESLIB	**- Directory for CERESlib (version - 19980109)**
PGSMSG	**- Directory which contains Toolkit and CERES Status Message Files**
PGSLIB	**- Directory which contains SGI 64-bit Toolkit library file**
F90COMP	**- SGI F90 compiler options (use the following options: -O2 -c -64 -OPT:fprop_limit=4000 -OPT:fold_arith_limit=3106)**
CERESHOME	**- Directory which is the CERES home directory**
PGSINC	**- Pointer to the PGS include file directory**
HDFDIR	**- Pointer to the HDF 4.1r1 home directory**

2. Change directory to the directory where you plan to install the Inversion Subsystems. (The following instructions assume that the directory will be **$CERESHOME**).

3. Uncompress and untar the tar files by typing the following commands:

```
> uncompress inversion_src_R2-45.tar.Z
> tar xf inversion_src_R2-45.tar
> uncompress inversion_anc_R2-45.tar.Z
> tar xf inversion_anc_R2-45.tar
> uncompress inversion_data_R2-45.tar.Z
> tar xf inversion_data_R2-45.tar
```

3.2 Compilation

1. **invsurf.exe**, the executable for Subsystems 4.5 and 4.6 Main Processor, is not provided on the tarfile. To create the executable on directory **$CERESHOME/inversion/bin/**, type the following commands:

   > **source $CERESENV**
   > **cd $CERESHOME/inversion/src/invsurf**
   > **make -f Makefile.inv**

2. **ssf2hdf.exe**, the executable for Subsystems 4.5 and 4.6 Postprocessor, is not provided on the tarfile. To create the executable on directory **$CERESHOME/inversion/bin/**, type the following commands:

   > **source $CERESENV**
   > **cd $CERESHOME/inversion/src/ssf2hdf**
   > **make -f Makefile.hdf**

4.0 Test and Evaluation Procedures

This section provides general information on how to execute Subsystems 4.5 and 4.6 and provides an overview of the test and evaluation procedures. It includes a description of what is being tested and the order in which the tests should be performed. There is a path in the Subsystem 4.5 code which executes a second Spectral Correction method. At the time of this delivery, this method will not be used in production runs and is not tested by these procedures.

4.1 Stand Alone Test Procedures

4.1.1 Main Processor PCF Generator

The Main Processor production script, run_4.5-6.1P1, references a Process Control File (PCF) which contains the correct file names and paths for the test procedures. This PCF for the test case is created by first executing an ASCII file generator, ascii_gen_4.5-6.1P1_test, and then executing the PCF generator, pcfgen_4.5-6.1P1.

For production runs, the ASCII file generator, ascii_gen_4.5-6.1P1, must be executed to create the ASCII input file for a particular production run. The ASCII file generator requires six command line arguments: 4-digit year, 2-digit month, 2-digit day, 2-digit hour-of-day, 6-digit configuration code, and production strategy. The PCF generator, pcfgen_4.5-6.1P1, is then executed using the newly created ASCII input file name as a command line argument.

1. Generate the ASCII input file for the test case:

   > cd $CERESHOME/inversion/bin
   > $CERESHOME/inversion/bin/ascii_gen_4.5-6.1P1_test 1997 12 28 00 000000 ValidationR1

   The following file will be generated in **$CERESHOME/inversion/bin/**:

   CER4.5-6.1P1_PCFin_TRMM-PFM-VIRS_ValidationR1_000000.1997122800

2. Generate the PCF for the test case:

   > $CERESHOME/inversion/bin/pcfgen_4.5-6.1P1 CER4.5-6.1P1_PCFin_TRMM-PFM-VIRS_ValidationR1_000000.1997122800

   The following PCF will be generated in **$CERESHOME/inversion/rcf/**:

   CER4.5-6.1P1_PCF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800

3. Compare the PCF for the test case with the PCF provided in this delivery, using the diff command:

   > diff $CERESHOME/inversion/rcf/CER4.5-6.1P1_PCF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800 $CERESHOME/inversion/data/out_exp/comp_data

The only differences between the two files should be in the directory paths where the tests were run.

4.1.2 Main Processor Execution

Execute the production script by typing the script name, run_4.5-6.1P1, followed by a string which designates the instance of the product. The string should be formatted, 'Sampling Strategy'_'Production Strategy'_'Configuration Code'.'Data Date'. The date parameter is formatted, YYYYMMDDHH, where YYYY is the 4-digit year, MM is the 2-digit month, DD is the 2-digit day and HH is the 2-digit hour of the data.

1. For the Main Processor test, use Sampling Strategy: TRMM-PFM-VIRS, Production Strategy: ValidationR1, Configuration Code: 000000 and Data Date: 1997122800

   > cd $CERESHOME/inversion/bin
   > $CERESHOME/inversion/bin/run_4.5-6.1P1 TRMM-PFM-VIRS_ValidationR1_000000.1997122800

   The Main Processor, Product Generation Executive (PGE) CER4.5-6.1P1, will be executed and will create:

   $CERESHOME/inversion/data/out_comp/data/CER_SSFB_TRMM-PFM-VIRS_ValidationR1_000000.1997122800
   $CERESHOME/inversion/data/out_comp/QC/CER_GQCA_TRMM-PFM-VIRS_ValidationR1_000000.1997122800
   $CERESHOME/inversion/data/out_comp/QC/CER_GQCI_TRMM-PFM-VIRS_ValidationR1_000000.1997122800
   $CERESHOME/inversion/data/out_comp/data/CER_SSFB_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met
   $CERESHOME/inversion/data/out_comp/QC/CER_GQCA_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met
   $CERESHOME/inversion/data/out_comp/QC/CER_GQCI_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met

   Main Processor Test Summary:

System Time:	1:53 minutes
Memory:	3664 K
Required Disk Space:	435 Megabytes

4.1.3 Postprocessor PCF Generator

The Postprocessor production script, run_4.5-6.2P1, also references a PCF which contains the correct file names and paths for the test procedures. The PCF for the test case is created by first executing an ASCII file generator, ascii_gen_4.5-6.2P1, and then executing the PCF generator, pcfgen_4.5-6.2P1.

For production runs, the ASCII file generator, ascii_gen_4.5-6.2P1 must be executed to create the ASCII input file for a particular production run. The ASCII file generator requires six command line arguments, 4-digit year, 2-digit month, 2-digit day, 2-digit hour of day, 6-digit configuration code, and production strategy. The PCF generator, pcfgen_4.5-6.2P1, is then executed using the newly created ASCII input file name as a command line argument.

1. Generate the ASCII input file for the test case:

   > cd $CERESHOME/inversion/bin
   > $CERESHOME/inversion/bin/ascii_gen_4.5-6.2P1 1997 12 28 00 000000 ValidationR1

   The following file will be generated in **$CERESHOME/inversion/bin/**:

   CER4.5-6.2P1_PCFin_TRMM-PFM-VIRS_ValidationR1_000000.1997122800

2. Generate the PCF for the test case:

   > $CERESHOME/inversion/bin/pcfgen_4.5-6.2P1 CER4.5-6.2P1_PCFin_TRMM-PFM-VIRS_ValidationR1_000000.1997122800

   The following PCF will be generated in **$CERESHOME/inversion/rcf/**:

   CER4.5-6.2P1_PCF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800

3. Compare the PCF for the test case, with the PCF provided in this delivery, using the diff command:

   > diff $CERESHOME/inversion/rcf/CER4.5-6.2P1_PCF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800 $CERESHOME/inversion/data/out_exp/comp_data

The only differences between the two files should be in the directory paths where the tests were run.

4.1.4 Postprocessor Execution

Execute the production script by typing the script name, run_4.5-6.2P1, followed by a string which designates the instance of the product. The string should be formatted, 'Sampling Strategy'_'Production Strategy'_'Configuration Code'.'Data Date'. The date parameter is formatted, YYYYMMDDHH, where YYYY is the 4-digit year, MM is the 2-digit month, DD is the 2-digit day and HH is the 2-digit hour of the data.

1. For the Postprocessor test, use Sampling Strategy: TRMM-PFM-VIRS, Production Strategy: ValidationR1, Configuration Code: 000000 and Data Date: 1997122800

   > cd $CERESHOME/inversion/bin
   > $CERESHOME/inversion/bin/run_4.5-6.2P1 TRMM-PFM-VIRS_ValidationR1_000000.1997122800

   The Postprocessor, PGE CER4.5-6.2P1, will be executed and will create:

   >$CERESHOME/inversion/data/out_comp/data/CER_SSF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800
   >$CERESHOME/inversion/data/out_comp/data/CER_SSF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met

   Postprocessor Test Summary:

System Time:	4:19 minutes
Memory:	16864 K
Required Disk Space:	360 Megabytes

4.2 Normal Operating Procedures

Before CERES Subsystems 4.5 and 4.6 can be executed in production, PCFs for PGE CER4.5-6.1P1 and PGE CER4.5-6.2P1 must be generated and the input data for the appropriate hour must be available from the successful execution of CERES Subsystems 4.4 and 12. The PRE_SSF, which is produced by the Convolution of Imager Cloud Properties with CERES Footprint Point Spread Function (PSF) Subsystem 4.4, and a MOA hourly product, which is produced by Subsystem 12, are required input. The procedures for generation of the PCFs for PGE CER4.5-6.1P1 and PGE CER4.5-6.2P1 are described in sections 4.1.1 and 4.1.3 of this document.

4.3 Evaluation Procedures

4.3.1 Exit Codes

All Subsystems 4.5 and 4.6 software terminates using the CERES defined EXIT CODES for the Langley TRMM Information System (LaTIS).

4.3.2 Log and Status File Results

The Error and Status Log Files, CER4.5-6.1P1_LogReport_TRMM-PFM-VIRS_ValidationR1_000000.1997122800 and CER4.5-6.2P1_LogReport_TRMM-PFM-VIRS_ValidationR1_000000.1997122800 are located in directory $CERESHOME/inversion/data/runlogs after CERES Subsystems 4.5 and 4.6 have been executed. Information contained in these log files can be compared to the expected contents of the Log Reports found in directory $CERESHOME/inversion/data/out_exp/comp_data.

4.3.3 Metadata Evaluation

Metadata files which end in extension, '.met', are located in the same directories as their corresponding output files after CERES Subsystems 4.5 and 4.6 PGEs have been executed. Metadata files, CER_GQCA_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met and CER_GQCI_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met, are written to directory, $CERESHOME/inversion/data/out_comp/QC. Metadata files, CER_SSFB_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met and CER_SSF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met, are written to directory, $CERESHOME/inversion/data/out_comp/data. The metadata contained in these files can be compared to the expected contents of the files with the same names found in directory $CERESHOME/inversion/data/out_exp/comp_data.

The only differences between the two files should be the production times and differences in the directory paths where the tests were run.

4.3.4 Execution of Comparison Software

The evaluation software for this Subsystem Main Processor will perform a single test. This test will copy all of the parameters that were written to the binary SSF by PGE CER4.5-6.1P1into a file and will compare those parameter values to the values in a comparison file provided with the software delivery.

1. The executable for the comparison software is not provided in the tar file. To create the executable on directory **$CERESHOME/inversion/test_suites/bin**, type the following commands:

   > **source $CERESENV**
   > **cd $CERESHOME/inversion/test_suites/src**
   > **make -f Makefile.cmp**

2. To execute the comparison software for the binary SSF, type the following commands:

   > **cd $CERESHOME/inversion/test_suites/bin**
   > **run_compare**

   Two files will be created:

$CERESHOME/inversion/data/out_comp/comp_data/SSF1.CMP
$CERESHOME/inversion/test_suites/results/CmpReport_1997122800

4.3.5 Evaluation of Comparison Software Output

This section provides the procedure for evaluating the output from Subsystems 4.5 and 4.6 comparison software.

1. Examine the comparison report files by typing:

   > more $CERESHOME/inversion/test_suites/results/ CmpReport_1997122800

   The final line of this file will report the status of the comparison between the generated data and the expected output.

2. File **$CERESHOME/inversion/test_suites/results/CmpReport_1997122800** will be automatically e-mailed to Sandy Nolan, s.k.nolan@larc.nasa.gov.

4.3.6 Evaluation of SSF HDF Product

This section provides the procedure for evaluating the output from the SSF HDF product produced by the test software. The HDF utility command, hdp, is available to compare the data on the HDF output file.

1. Execute the following HDF utility command:

   > hdp dumpsds $CERESHOME/inversion/data/out_comp/data/ CER_SSF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800 > dump.hdf

2. Compare file dump.hdf with the HDF output dump, dump.hdf, found in **$CERESHOME/ inversion/data/out_exp/comp_data.**

The only differences between these two files should be the names of the files on which the 'hdp' command was performed.

4.4 Solutions to Possible Problems

1. All output files are opened with Status = NEW in Subsystem 4.5 and 4.6 software. These files must be removed before rerunning these test procedures.
2. Environment variable, F90COMP, must be set to the following SGI compiler options:

   setenv F90COMP '-O2 -c -64 -OPT:fprop_limit=4000 -OPT:fold_arith_limit=3106'

   Compiler warnings will be generated and runtime problems may occur if these SGI compiler options are not set.
3. When running the production scripts, run_4.5-6.1P1 and run_4.5-6.2P1, the system message, 'No match', may be written to the screen. This message occurs when the scripts try to remove an old output file that does not exist. This does not signify a problem.
4. Use CERESlib Version 19980109 or later.

APPENDIX A
Acronyms and Abbreviations

Appendix A
Acronyms and Abbreviations

ADM	Angular Distribution Models
ASCII	American Standard Code Information Interchange
CERES	Clouds and the Earth's Radiant Energy System
CERESlib	CERES library
DAAC	Distributed Active Archive Center
EOS	Earth Observing System
EOS-AM	EOS Morning Crossing Mission
EOS-PM	EOS Afternoon Crossing Mission
ERBE	Earth Radiation Budget Experiment
ERBS	Earth Radiation Budget Satellite
F90	Fortran 90
HDF	Hierarchical Data Format
LaTIS	Langley TRMM Information System
LW	Longwave
MOA	Meteorological, Ozone, and Aerosol
NASA	National Aeronautics and Space Administration
NOAA	National Oceanic and Atmospheric Administration
PCF	Process Control File
PGE	Product Generation Executives
PSF	Point Spread Function
PRE_SSF	Preliminary Single Satellite CERES Footprint TOA and Surface Fluxes
QC	Quality Control
SMF	Status Message File
SSF	Single Satellite CERES Footprint TOA and Surface Fluxes, Clouds
SW	Shortwave
TOA	Top-of-Atmosphere
TRMM	Tropical Rainfall Measuring Mission

APPENDIX B
Directory Structure Diagram

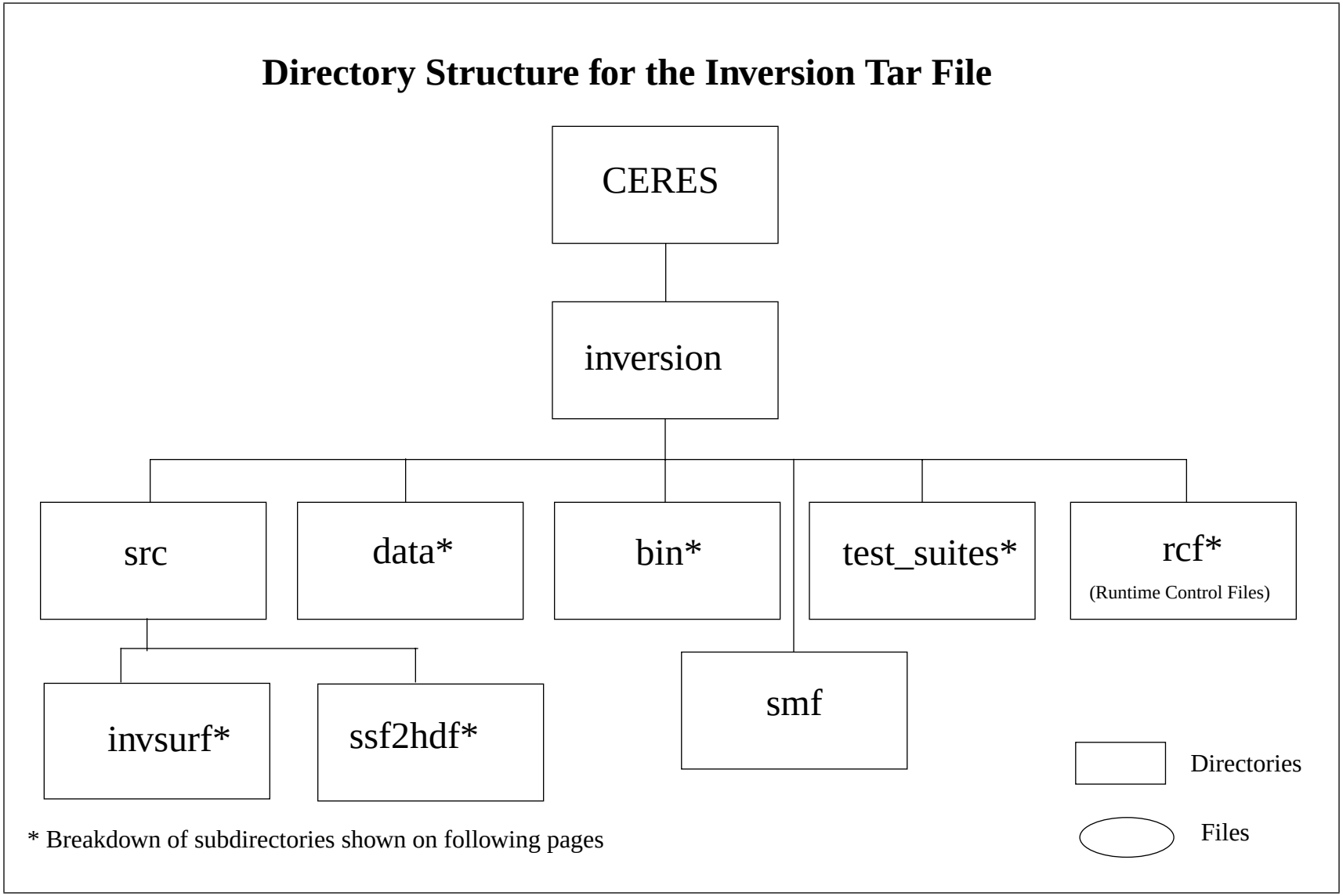


Figure B-1. Directory Structure for the Inversion Tar File (1 of 17)

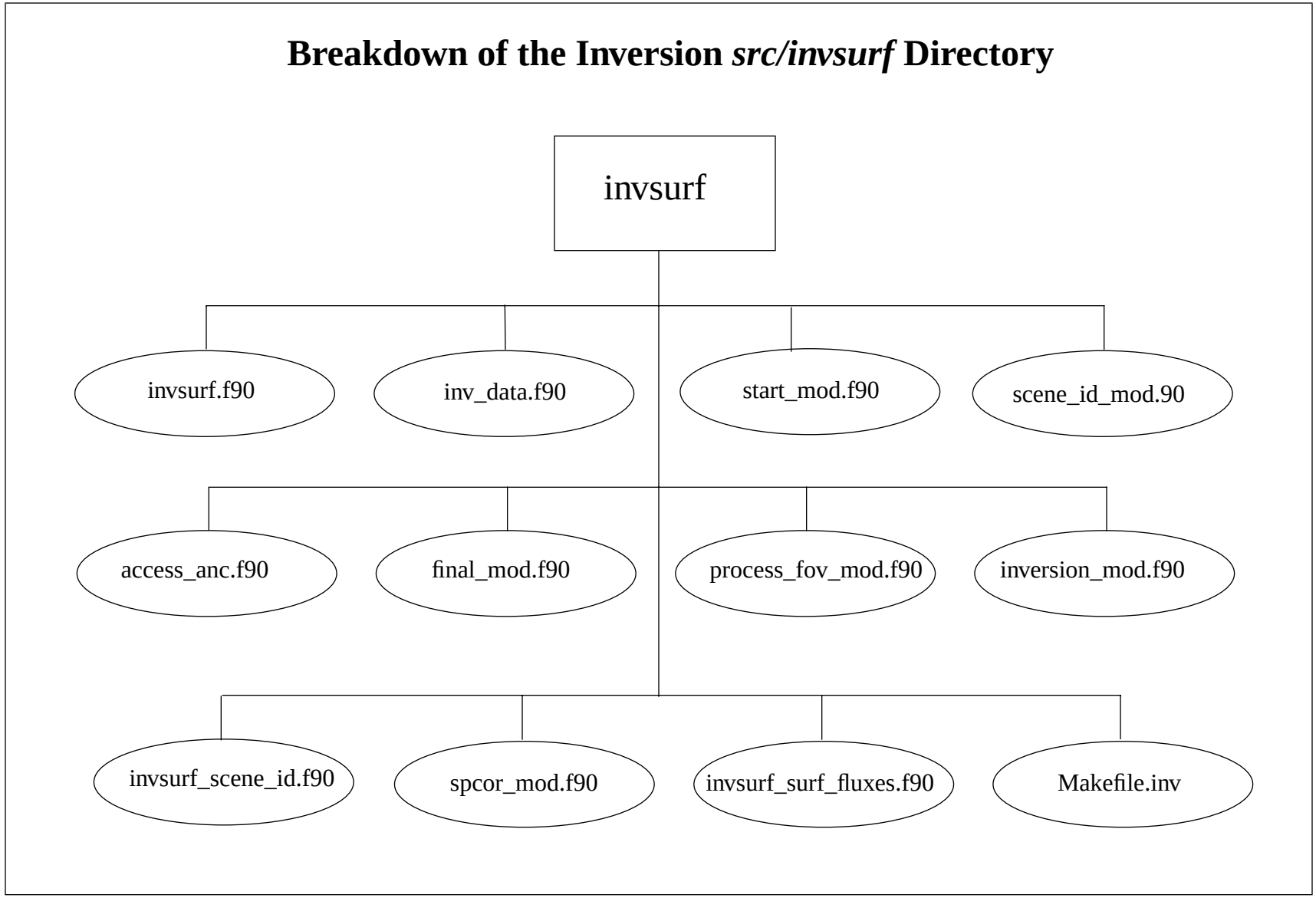


Figure B-1. Directory Structure for the Inversion Tar File (2 of 17)

Breakdown of the Inversion *src/ssf2hdf* Directory

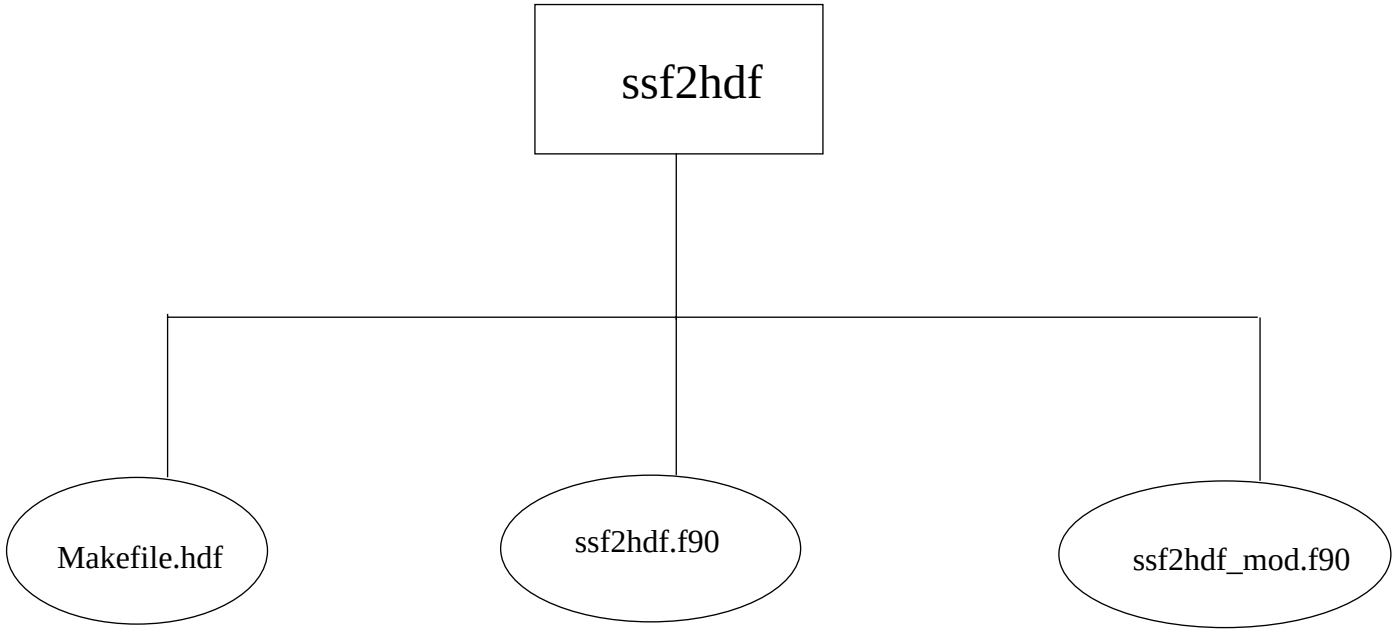
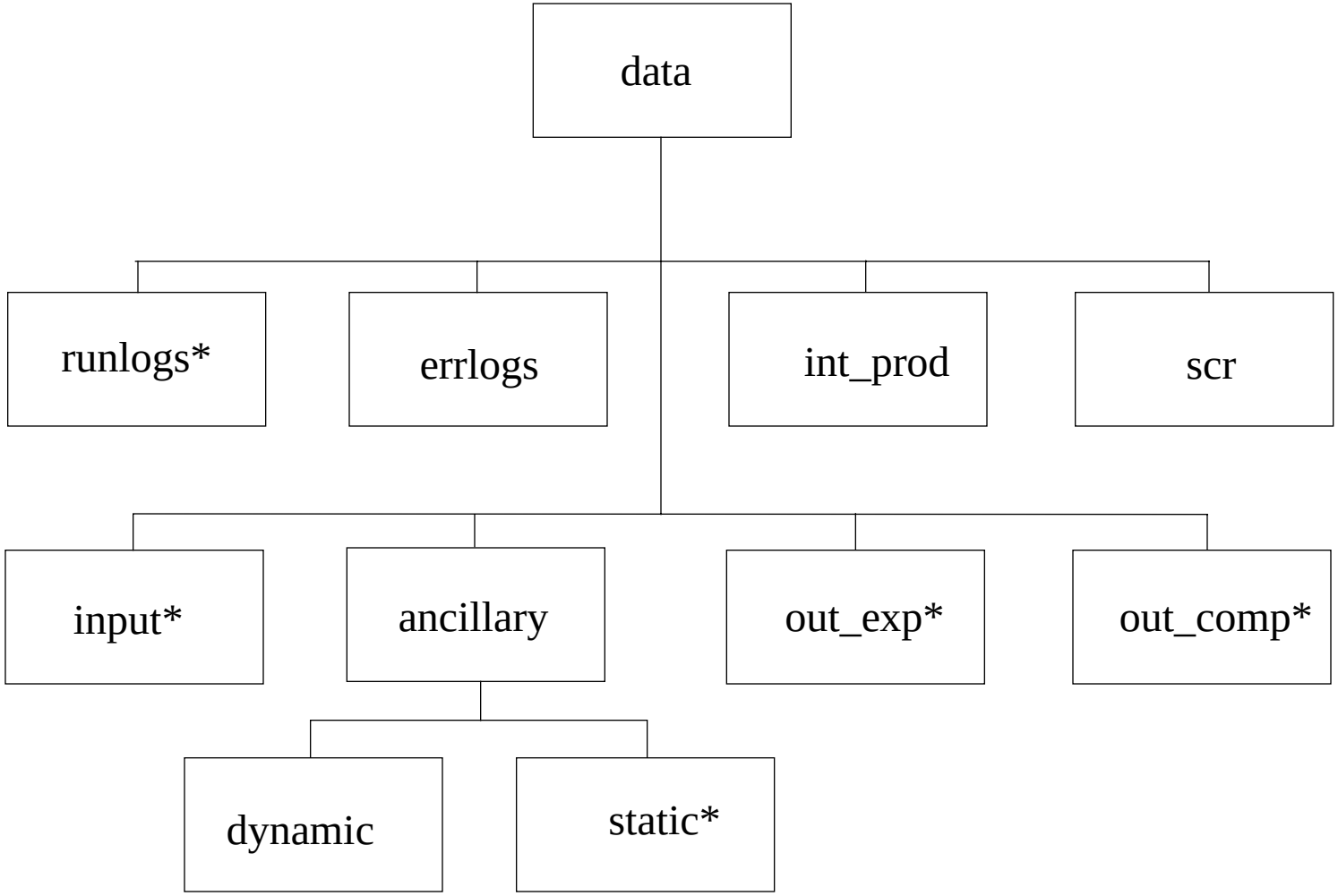


Figure B-1. Directory Structure for the Inversion Tar File (3 of 17)

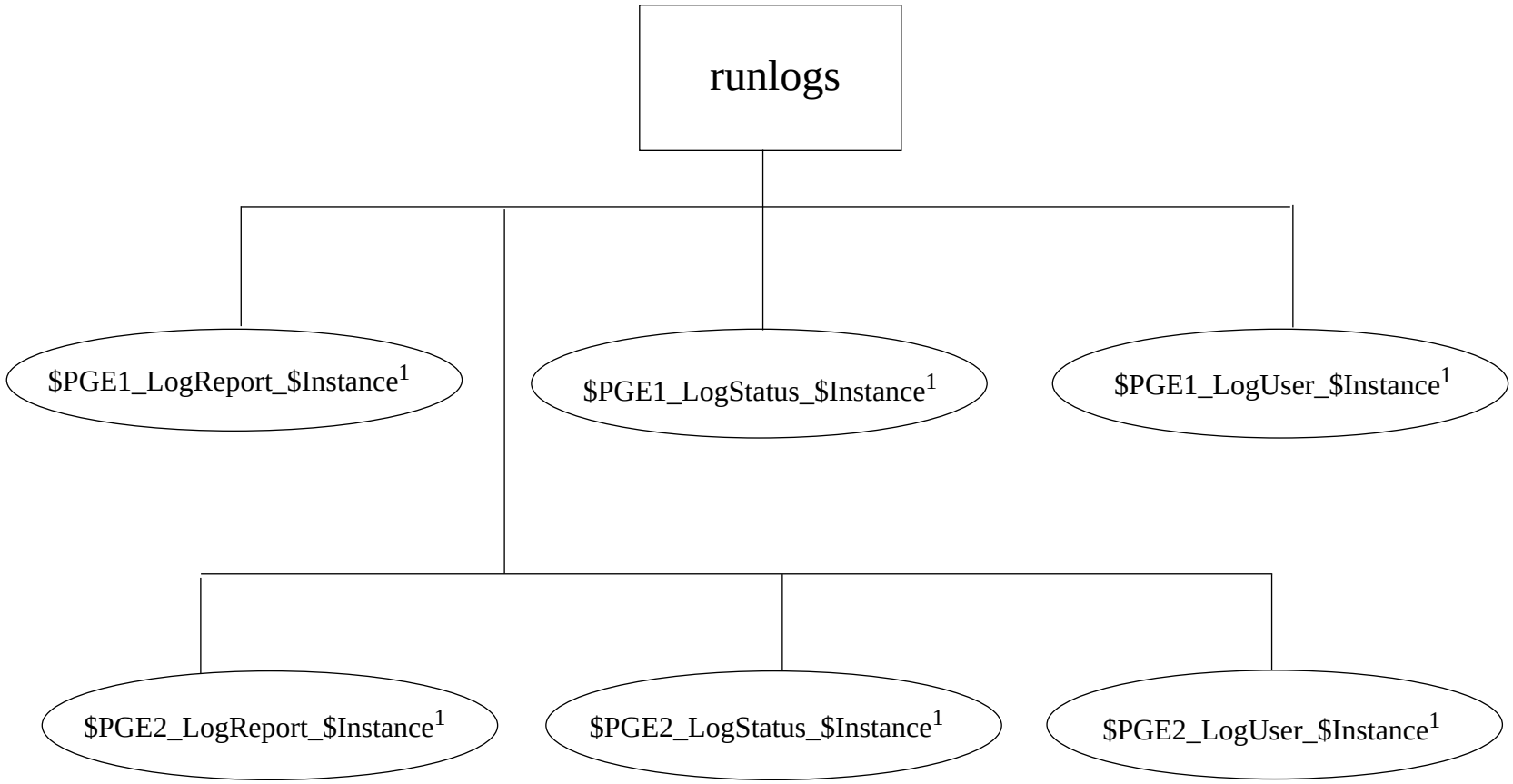
Breakdown of the Inversion *data* Directory



* Breakdown of subdirectories shown on following pages

Figure B-1. Directory Structure for the Inversion Tar File (4 of 17)

Breakdown of the Inversion *data/runlogs* Directory



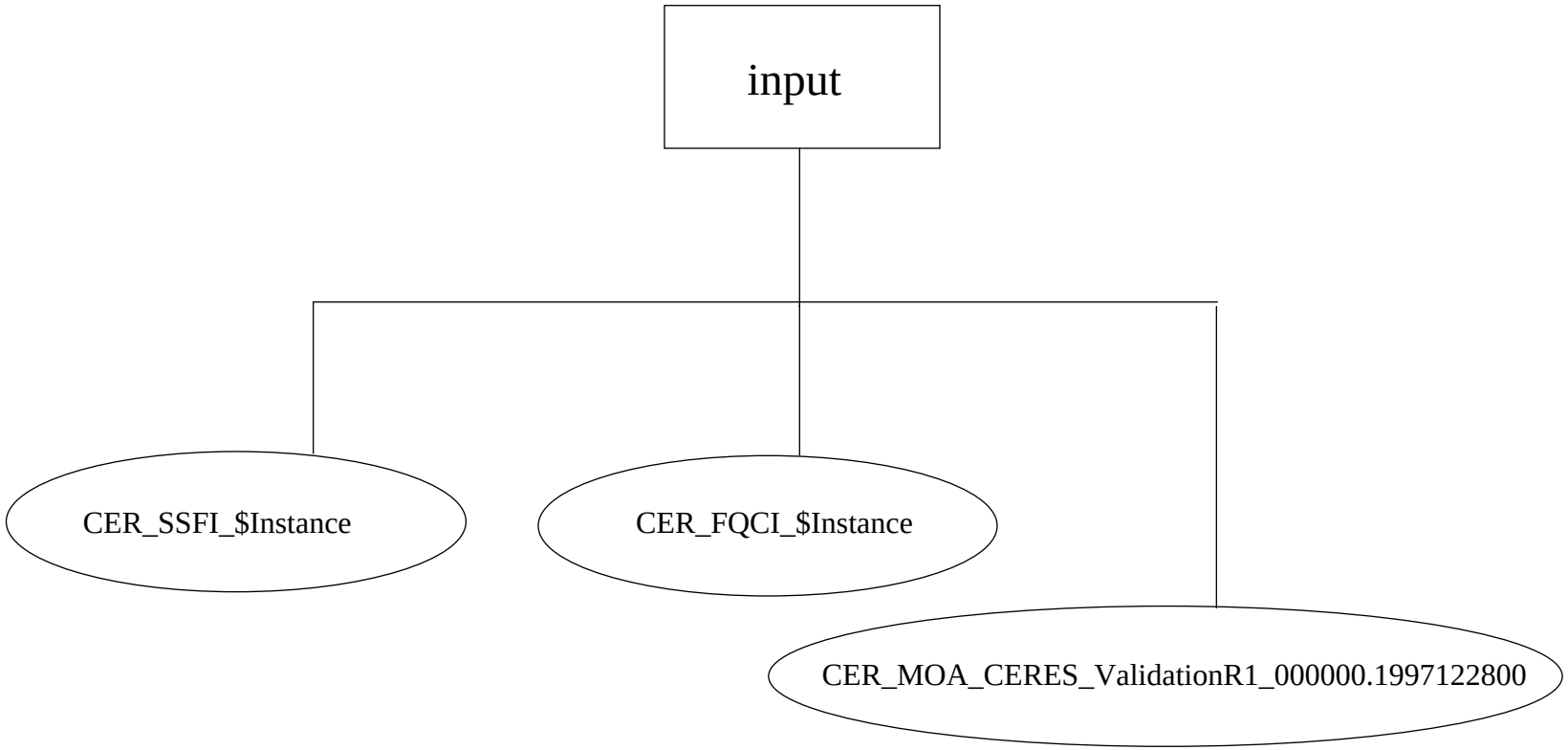
Note: $Instance = ‘TRMM-PFM-VIRS_ValidationR1_000000.1997122800’

Note: $PGE1 = ‘CER4.5-6.1P1’ and $PGE2 = ‘CER4.5-6.2P1’

[1]These files are not included in the tar file but will be produced by the Subsystem software.

Figure B-1. Directory Structure for the Inversion Tar File (5 of 17)

Breakdown of the Inversion *data/input* Directory



Note: $Instance = ‘TRMM-PFM-VIRS_ValidationR1_000000.1997122800’

Figure B-1. Directory Structure for the Inversion Tar File (6 of 17)

Breakdown of the Inversion *data/ancillary/static* Directory

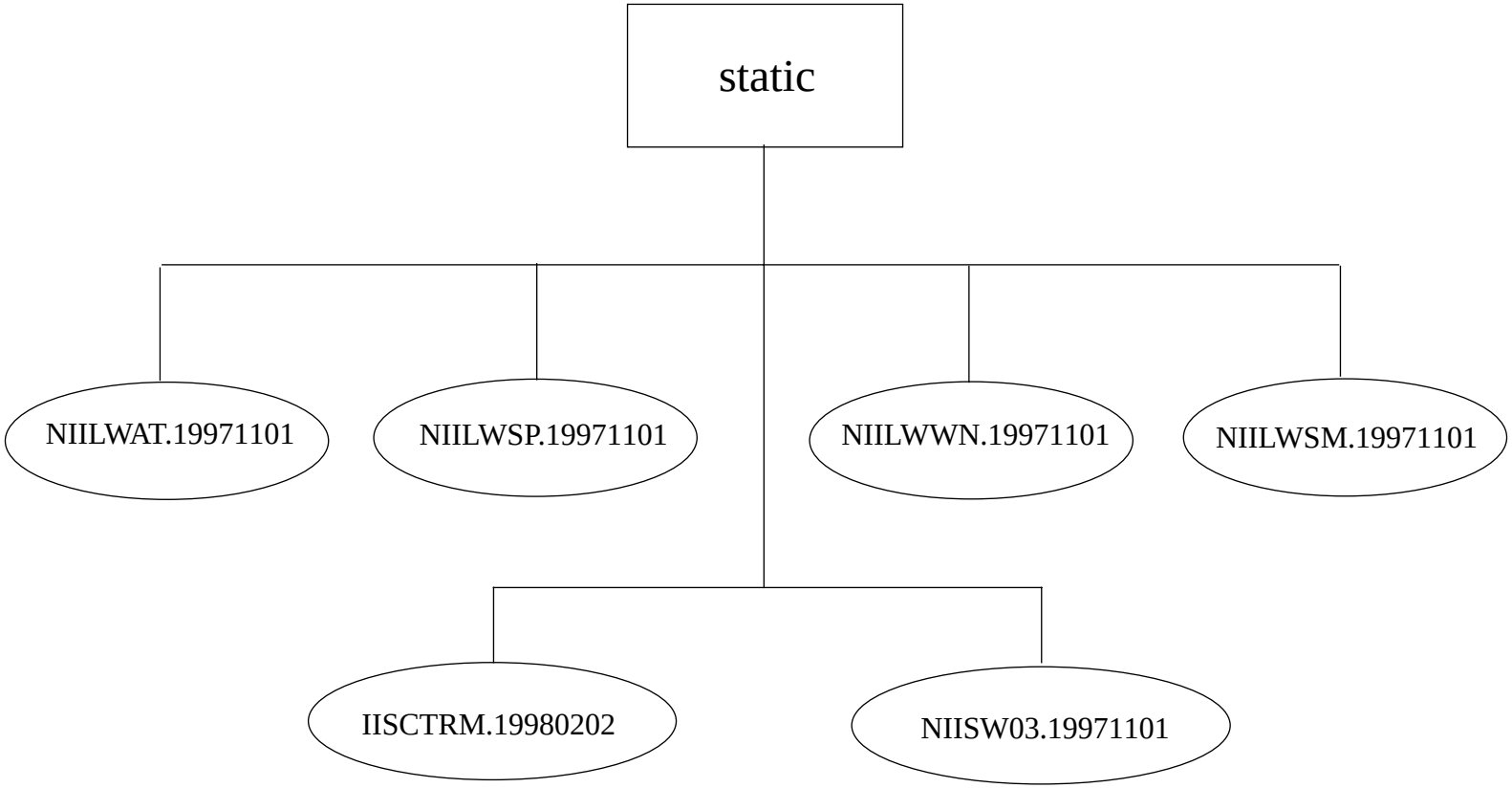
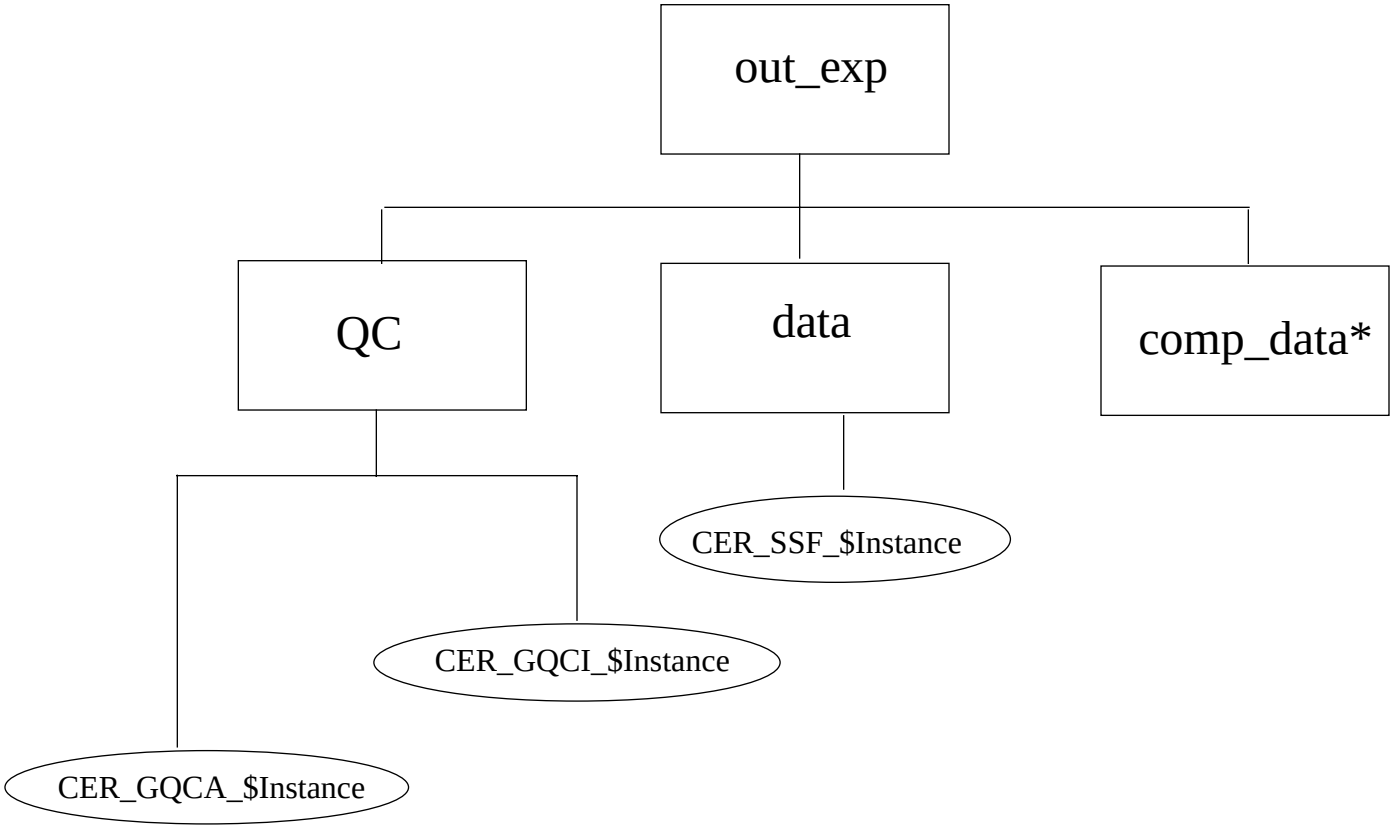


Figure B-1. Directory Structure for the Inversion Tar File (7 of 17)

Breakdown of the Inversion *data/out_exp* Directory



Note: $Instance = 'TRMM-PFM-VIRS_ValidationR1_000000.1997122800'

* Breakdown of subdirectory shown on following page

Figure B-1. Directory Structure for the Inversion Tar File (8 of 17)

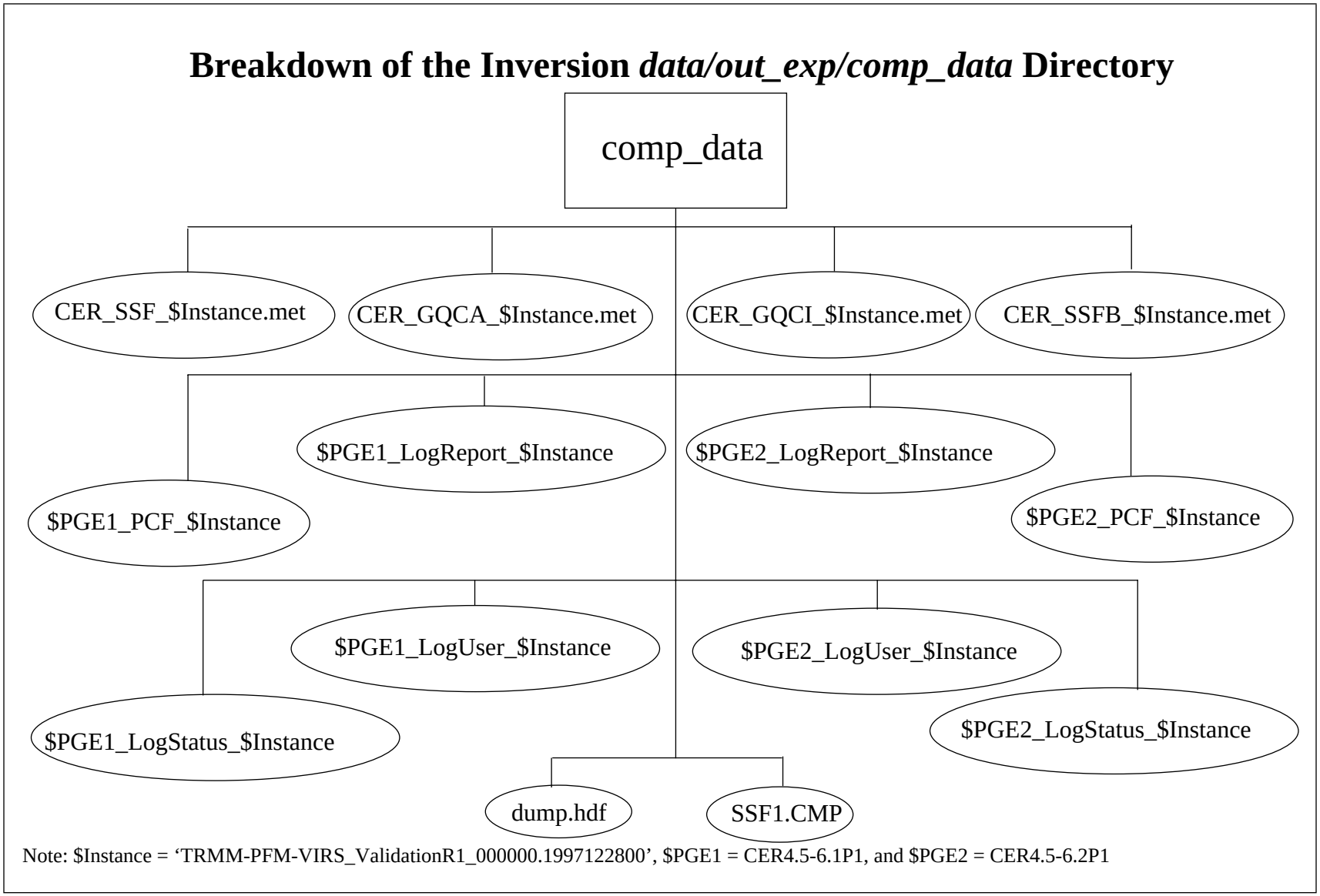
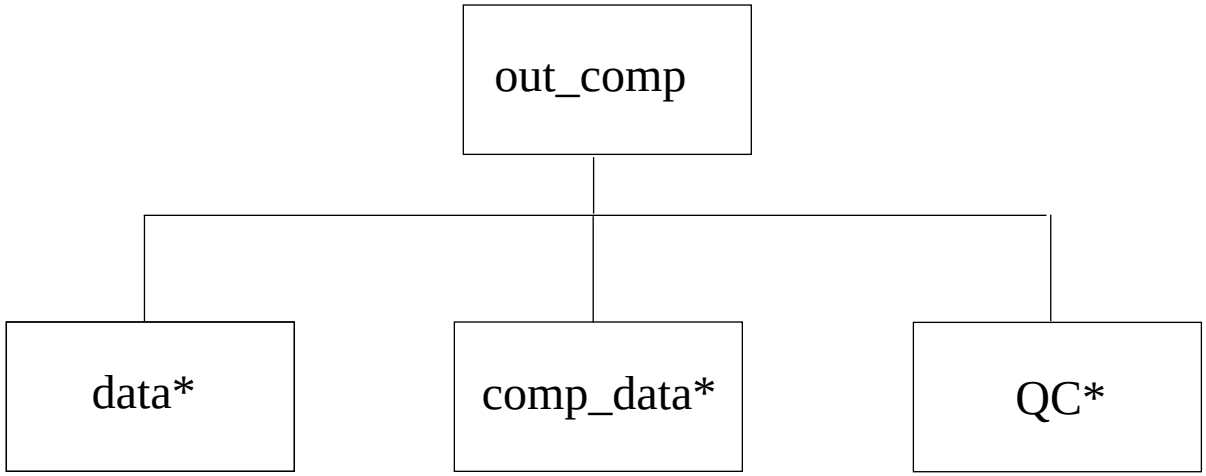


Figure B-1. Directory Structure for the Inversion Tar File (9 of 17)

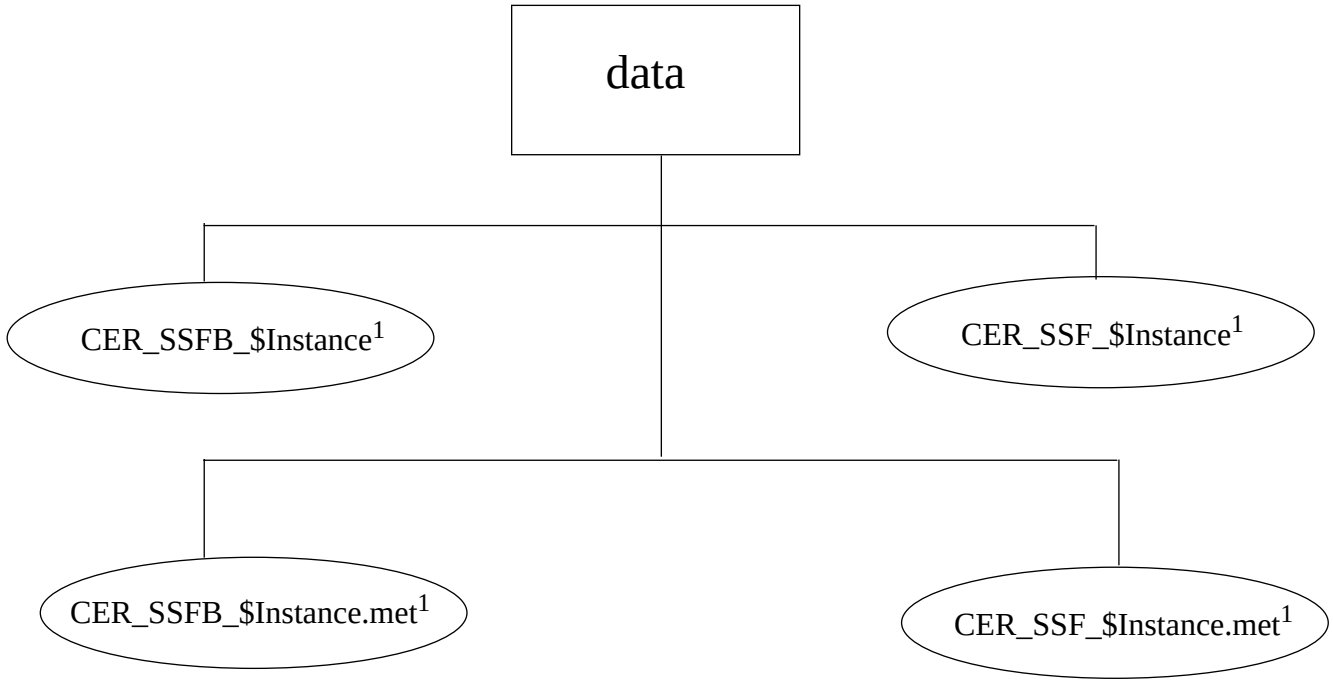
Breakdown of the Inversion *data/out_comp* Directory



* Breakdown of subdirectories shown on following pages

Figure B-1. Directory Structure for the Inversion Tar File (10 of 17)

Breakdown of the Inversion *data/out_comp/data* Directory

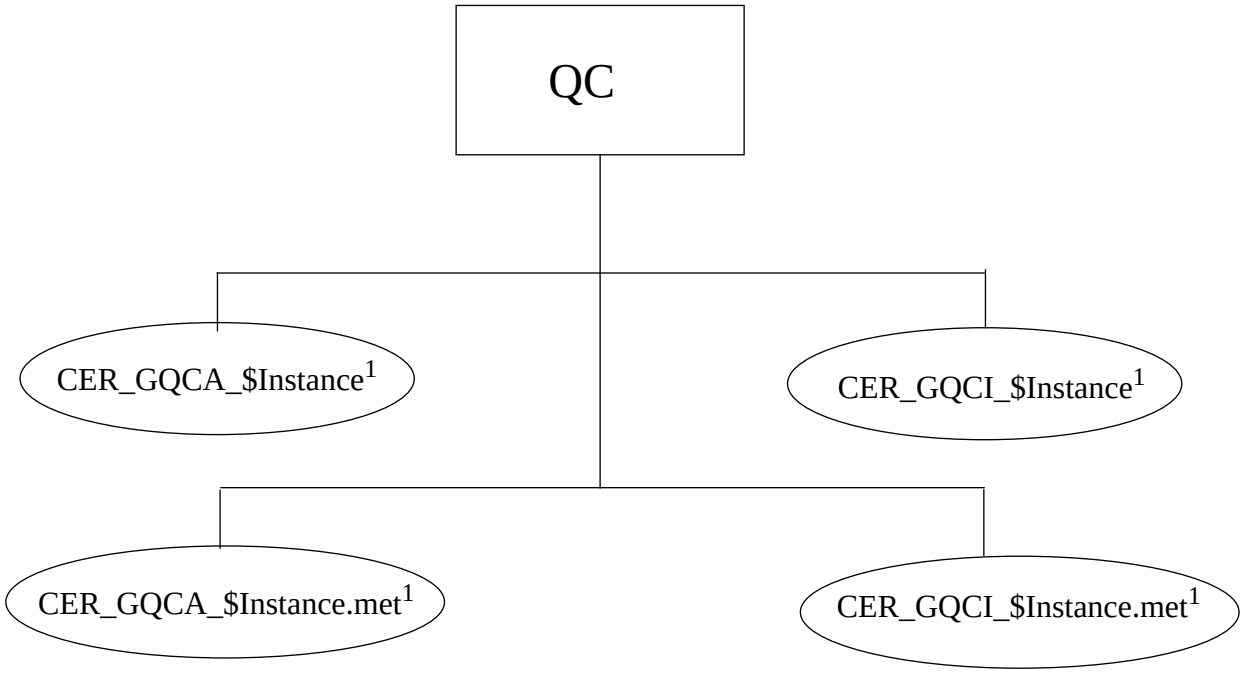


Note: $Instance = ‘TRMM-PFM-VIRS_ValidationR1_000000.1997122800’

[1]These files are not included in the tar file but will be produced by the Subsystem software.

Figure B-1. Directory Structure for the Inversion Tar File (11 of 17)

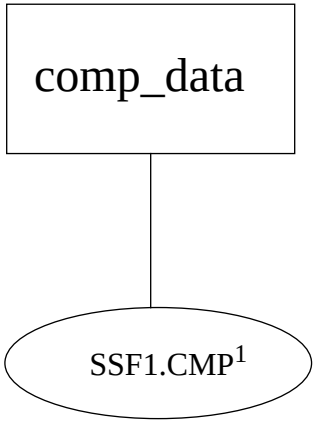
Breakdown of the Inversion *data/out_comp/QC* Directory



Note: $Instance = ‘TRMM-PFM-VIRS_ValidationR1_000000.1997122800’

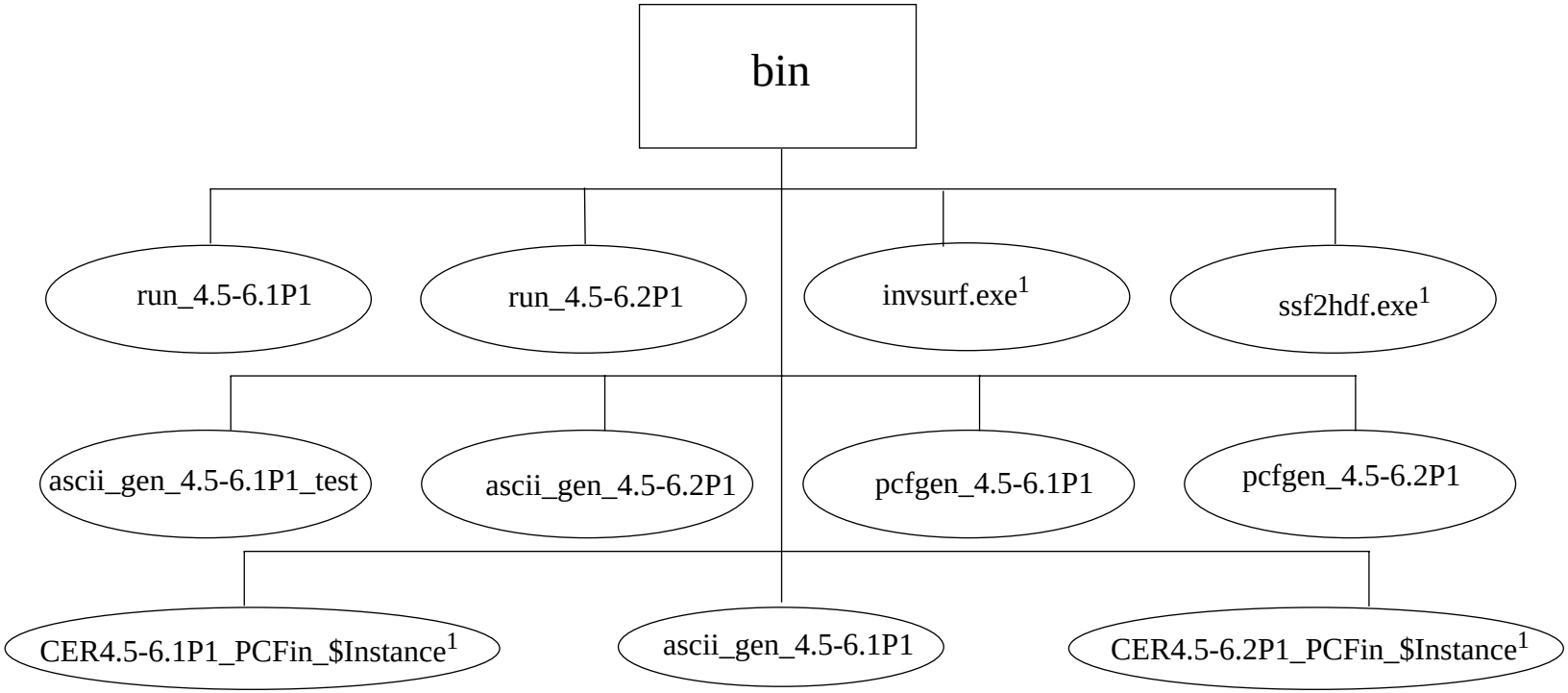
[1]These files are not included in the tar file but will be produced by the Subsystem software.

Breakdown of the Inversion *data/out_comp/comp_data* Directory



[1]These files are not included in the tar file but will be produced by the Subsystem software.

Breakdown of the Inversion *bin* Directory

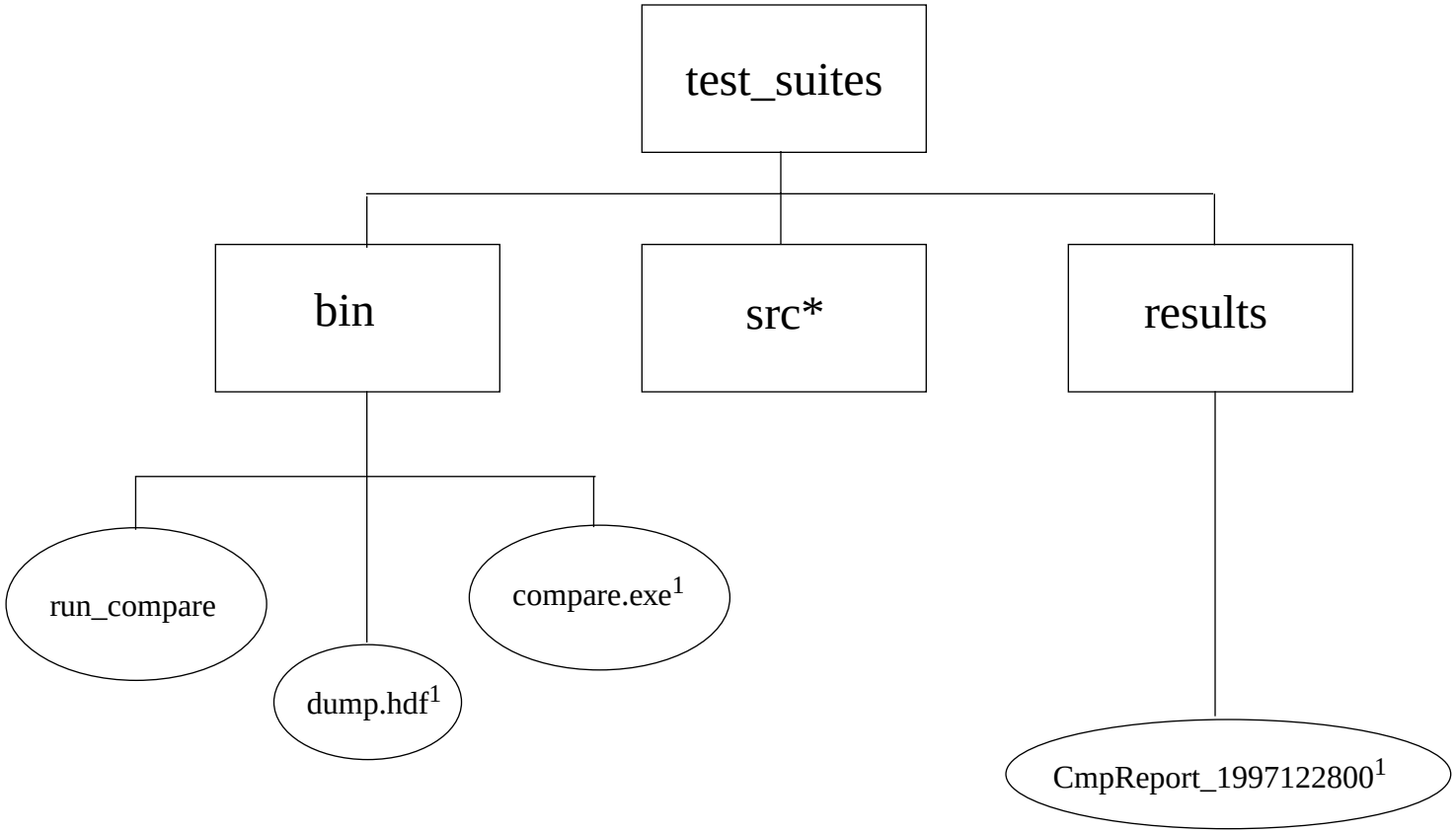


Note: $Instance = ‘TRMM-PFM-VIRS_ValidationR1_000000.1997122800’

[1]These files are not included in the tar file but will be produced by the Subsystem software.

Figure B-1. Directory Structure for the Inversion Tar File (14 of 17)

Breakdown of the Inversion *test_suites* Directory



* Breakdown of subdirectory shown on following page

[1]These files are not included in the tar file but will be produced by the Subsystem software.

Figure B-1. Directory Structure for the Inversion Tar File (15 of 17)

Breakdown of the Inversion *test_suites/src* Directory

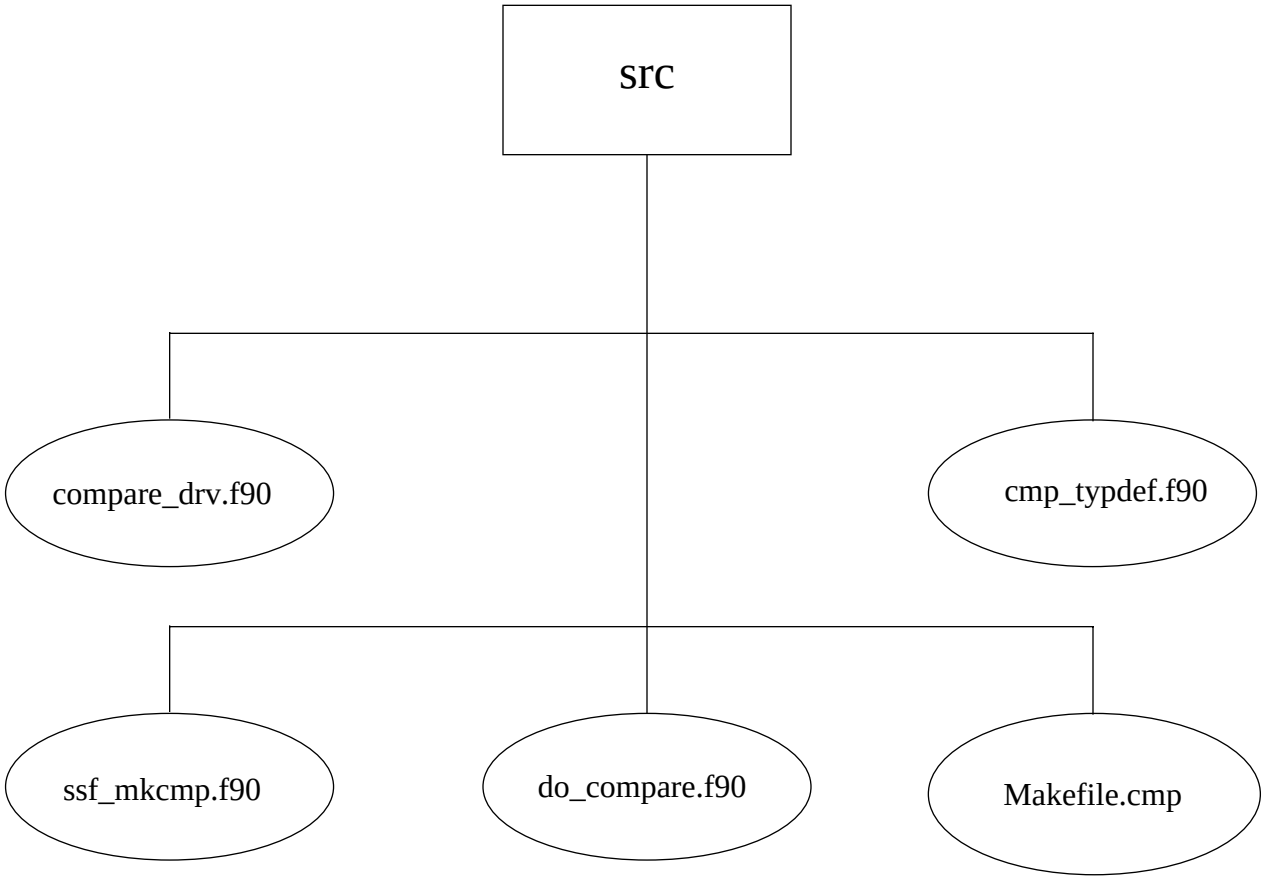
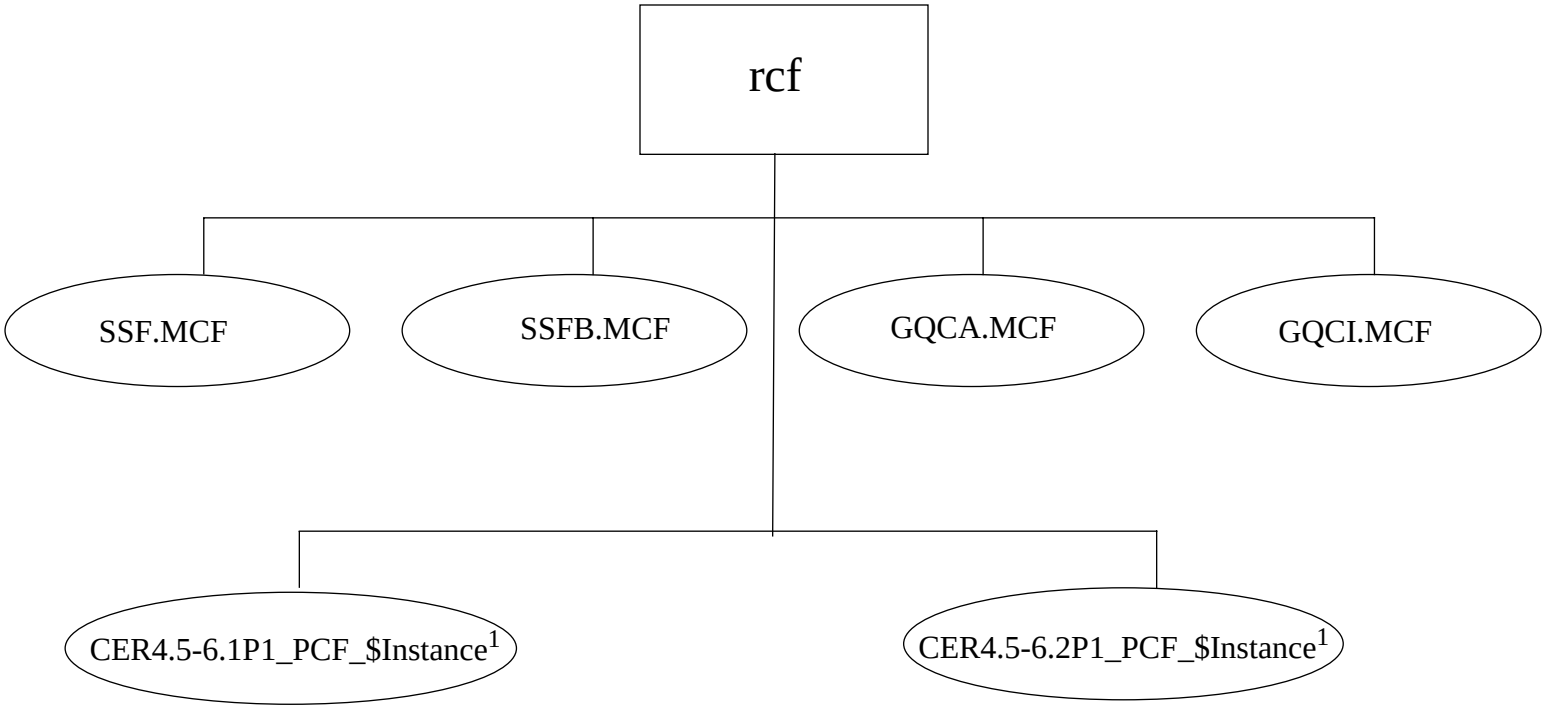


Figure B-1. Directory Structure for the Inversion Tar File (16 of 17)

Breakdown of the Inversion *rcf* Directory



Note: $Instance = 'TRMM-PFM-VIRS_ValidationR1_000000.1997122800'

[1]These files are not included in the tar file but will be produced by the Subsystem software.

Figure B-1. Directory Structure for the Inversion Tar File (17 of 17)

APPENDIX C
File Description Tables

Appendix C
File Description Tables

C.1 Production Scripts and Executables

Table C.1-1. Production Scripts and Executables

File Name	Format	Description
ascii_gen_4.5-6.1P1_test	ASCII	C-Shell script which creates the PCF generator's ASCII file needed by the Main Processor for the test case
ascii_gen_4.5-6.1P1	ASCII	C-Shell script which creates the PCF generator's ASCII file needed by the Main Processor
pcfgen_4.5-6.1P1	ASCII	C-Shell script which creates the PCF for the Main Processor
run_4.5-6.1P1	ASCII	C-Shell script which executes the Main Processor
invsurf.exe[1]	Binary	Main Processor executable
ascii_gen_4.5-6.2P1	ASCII	C-Shell script which creates the PCF generator's ASCII file needed by the Post Processor
pcfgen_4.5-6.2P1	ASCII	C-Shell script which creates the PCF for the Post Processor
run_4.5-6.2P1	ASCII	C-Shell script which executes the Post Processor
ssf2hdf.exe[1]	Binary	Post Processor executable

[1]These files will be generated on execution of Subsystem software and are not included in the tar file.

C.2 Processing Control Files (PCF), Metadata Control Files (MCF) and Status Message Files(SMF)

The Process Control Files are not included in the Software Delivery Package. They will be created by the PCF generator scripts. Subsystem 4.5 and 4.6 software does not create any Status Message Files.

Table C.2-1. Metadata Control Files

File Name	Format	Description
GQCA.MCF	ODL	MCF for ASCII QC Report for Main Processor
GQCI.MCF	ODL	MCF for Binary QC Report for Main Processor
SSF.MCF	ODL	MCF for SSF's HDF file for Post Processor
SSFB.MCF	ODL	MCF for Binary SSF file for Main Processor

Table C.2-2. Process Control Files

File Name	Format	Description
CER4.5-6.1P1_PCF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800[1]	ASCII	Process Control File template for Main Processor
CER4.5-6.2P1_PCF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800[1]	ASCII	Process Control File template for Post Processor

[1]These files will be generated on execution of Subsystem software and are not included in the tar file.

C.3 Production Makefiles

Table C.3-1. Production Makefiles

File Name	Format	Description
Makefile.inv	ASCII	Makefile to Produce Main Processor Executable
Makefile.hdf	ASCII	Makefile to Produce Post Processor Executable

C.4 Ancillary Input Data

Table C.4-1. Ancillary Input Data

File Name	Format	Description
NIISW03.19971101	Binary	SW Angular Distribution Models (ADM)
NIILWAT.19971101	Binary	LW ERBE ADM for Sep., Oct., Nov.
NIILWSP.19971101	Binary	LW ERBE ADM for Mar., Apr., May
NIILWSM.19971101	Binary	LW ERBE ADM for Jun., Jul., Aug.
NIILWWN.19971101	Binary	LW ERBE ADM for Dec., Jan., Feb.
IISCTRM.19980202	Binary	Spectral Correction Coefficients for TRMM

C.5 Primary Input Data

The "Test Data" files included in the Software Delivery Package are listed in Table C.5-1.

Table C.5-1. Primary Input Data

File Name	Format	Description
CER_SSFI_TRMM-PFM-VIRS_ValidationR1_000000.1997122800	Binary	Interim SSF file produced by Subsystem 4.4
CER_MOA_CERES_ValidationR1_000000.1997122800	Binary	MOA product produced by Subsystem 12
CER_FQCI_TRMM-PFM-VIRS_ValidationR1_000000.1997122800	Binary	Interim SSFQC file produced by Subsystem 4.4
CER_SSFB_TRMM-PFM-VIRS_ValidationR1_000000.1997122800[1]	Binary	SSF file produced by the Main Processor and used as input to the Post Processor

[1]This file will be generated on execution of Subsystem software and are not included in the tar file.

C.6 Output Data Files (Expected Results)

Table C.6-1. Expected Output Data

File Name	Format	Description
CER_GQCA_TRMM-PFM-VIRS_ValidationR1_000000.1997122800	ASCII	SSF QC report
CER_GQCI_TRMM-PFM-VIRS_ValidationR1_000000.1997122800	Binary	SSF Binary QC file
CER_SSF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800	HDF	SSF HDF file
CER4.5-6.1P1_PCF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800	ASCII	Process Control File for Main Processor
CER4.5-6.2P1_PCF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800	ASCII	Process Control File for Post Processor
CER_GQCA_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met	ODL	Metadata File for ASCII QC Report
CER_GQCI_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met	ODL	Metadata File for Binary QC Report
CER_SSF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met	ODL	Metadata File for SSF's HDF File
CER_SSFB_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met	ODL	Metadata File for Binary SSF File

C.7 Output Data Files (Production Results)

Table C.7-1. Production Output Data

File Name	Format	Description
CER_GQCA_TRMM-PFM-VIRS_ValidationR1_000000.1997122800[1]	ASCII	ASCII SSF QC report
CER_GQCI_TRMM-PFM-VIRS_ValidationR1_000000.1997122800[1]	Binary	Binary SSF QC file
CER_SSFB_TRMM-PFM-VIRS_ValidationR1_000000.1997122800[1]	Binary	Binary SSF
CER_SSF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800[1]	HDF	SSF in HDF format
CER_GQCA_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met[1]	ODL	Metadata File for ASCII QC Report
CER_GQCI_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met[1]	ODL	Metadata File for Binary QC Report
CER_SSF_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met[1]	ODL	Metadata File for SSF's HDF File
CER_SSFB_TRMM-PFM-VIRS_ValidationR1_000000.1997122800.met[1]	ODL	Metadata File for Binary SSF File
CER4.5-6.1P1_LogStatus_TRMM-PFM-VIRS_ValidationR1_000000.1997122800[1]	ASCII	Status Log File for Main Processor
CER4.5-6.1P1_LogReport_TRMM-PFM-VIRS_ValidationR1_000000.1997122800[1]	ASCII	Report Log File for Main Processor
CER4.5-6.1P1_LogUser_TRMM-PFM-VIRS_ValidationR1_000000.1997122800[1]	ASCII	User Log File for Main Processor
CER4.5-6.2P1_LogStatus_TRMM-PFM-VIRS_ValidationR1_000000.1997122800[1]	ASCII	Status Log File for Post Processor
CER4.5-6.2P1_LogReport_TRMM-PFM-VIRS_ValidationR1_000000.1997122800[1]	ASCII	Report Log File for Post Processor
CER4.5-6.2P1_LogUser_TRMM-PFM-VIRS_ValidationR1_000000.1997122800[1]	ASCII	User Log File for Post Processor

[1]These files will be generated on execution of Subsystem software and are not included in the tar file.

C.8 Output Temporary Data Files (Production Results)

Table C.8-1. Output Temporary Data Files

File Name	Format	Description
CER4.5-6.1P1_PCFin_TRMM-PFM-VIRS_ValidationR1_000000.1997122800[1]	ASCII	ASCII file created by the ASCII file generator to be used by the Main Processor's PCF generator
CER4.5-6.2P1_PCFin_TRMM-PFM-VIRS_ValidationR1_000000.1997122800[1]	ASCII	ASCII file created by the ASCII file generator to be used by the Post Processor's PCF generator

[1]These files will be generated on execution of Subsystem software and are not included in the tar file.

C.9 Error and Status Message Files (Expected Results)

Table C.9-1. Error and Status Message Files

File Name	Format	Description
CER4.5-6.1P1_LogStatus_TRMM-PFM-VIRS_ValidationR1_000000.1997122800	ASCII	Status Log File for Main Processor
CER4.5-6.1P1_LogReport_TRMM-PFM-VIRS_ValidationR1_000000.1997122800	ASCII	Report Log File for Main Processor
CER4.5-6.1P1_LogUser_TRMM-PFM-VIRS_ValidationR1_000000.1997122800	ASCII	User Log File for Main Processor
CER4.5-6.2P1_LogStatus_TRMM-PFM-VIRS_ValidationR1_000000.1997122800	ASCII	Status Log File for Post Processor
CER4.5-6.2P1_LogUser_TRMM-PFM-VIRS_ValidationR1_000000.1997122800	ASCII	Report Log File for Post Processor
CER4.5-6.2P1_LogUser_TRMM-PFM-VIRS_ValidationR1_000000.1997122800	ASCII	User Log File for Post Processor

C.10 Test Evaluation Software

Table C.10-1. Test Evaluation Scripts and Executables

File Name	Format	Description
run_compare	ASCII	Script for comparing binary SSFdata
compare.exe[1]	Binary	Comparison executable

[1]This file will be generated on execution of Subsystem software and are not included in the tar file.

Table C.10-2. Test Evaluation Source Code and Makefile

File Name	Format	Description
compare_drv.f90	ASCII	Drives comparison software
ssf_mkcmp.f90	ASCII	Creates comparison file
do_compare.f90	ASCII	Compares two comparison files
cmp_typdef.f90	ASCII	Module containing type definition for comparison data structure
Makefile.cmp	ASCII	Makefile for comparison software

Table C.10-3. Test Evaluation Output (Expected Results)

File Name	Format	Description
SSF1.CMP	Binary	SSF binary comparison file
dump.hdp	ASCII	SSF HDF comparison file

Table C.10-4. Test Evaluation Output (Test Results)

File Name	Format	Description
SSF1.CMP[1]	Binary	SSF binary comparison file
dump.hdp[1]	ASCII	File produced by the "hdp" utility to compare test data to expected data
CmpReport_1997122800[1]	ASCII	Binary SSF comparison report

[1]These files will be generated on execution of Subsystem software and are not included in the tar file.