



International Satellite Cloud Climatology Project

Operations Guide Build 5

CDR Program Document Number: CDRP-MAN-0876
Configuration Item Number: 01B-29
Revision 0 / May 16, 2017

REVISIONS from Build 4 Ops Guide

Section	Change
1.1.1	Add calibration and comparison_plots to directory structure
1.1.2	Add reference to Ancillary Production Guide
1.2.4	Note staging requirements
1.3.6	Project dates for pre-1983 processing
1.3.7	Clarify timeslot descriptions and duplicate orbit handling
1.3.10	Clarify HGS metadata handling
2	Modify data flow diagram to include snowice input to HGG Add process flowcharts for RE-processing and NEW-processing
3	Expand descriptions of production monitoring processes and production plot interpretation guidelines, add comparison plots.
3.1	Illustrate use of scheduling GUI for reviewing sathist info
4.6.1	Note staging requirement for b4prod
4.6.5	Add definition of coast pixels excluded from HXS
4.7.3	Snowice, atmosphere, ozone always reported, even in empty cells
4.7.5	Note staging requirement for d1fill
4.7.5.5	HGG cell origin applies only to filled vars
4.7.5.7	Snowice, atmosphere, ozone always reported, not subject to filling
4.7.6	Snowice, atmosphere, ozone always reported in monthly means
4.8	Note staging requirement for dxgfill
4.9.1	Add calcoefs_update script, add initial HBT script
4.9.2	Add NetCDF bug ticket description for CDL files w/comment field
4.9.3	Add reference to btcacor, sattable_module, topo_module, add sat_setup, modify tables subdir description, add initial HBT programs, add solzen module
4.10.3	Add reference to topo_module
4.12	Modify PTRANS and REFGEO prd/tables/* description. Add reference to SATTABLE.ALL
5.1	Clarify satname/satpos/spc interactions in sathist
5.2	Unused radius = 0, not N/A, add solcon field definition, clarify Sat.pm key value assignment rules
5.2.2	Add references to radius, refgeo sections, clarify ptrans, add initial calib
5.4	Clarify subsatlon usage in sathier
5.5	Clarify RADIUS optimal value definition
5.7	Add PTRANS file creation procedure
5.8	Add REFGEO file creation procedure
5.9	Add Radius computation procedure
5.10	Add reference to calibration setup procedures for 2010 forward
6.1	Revise array dimensions for B1QC output
6.3	Clarify index for timeslots in QCresults files
6.4	Add section describing QC plot interpretation
6.5	Add section describing QC visual inspection procedure
7	Add section 7 for initial calibration setup procedures for new satellites
8	Add section 8 for calibration monitoring and update procedures (under development)
9	RECAL section renumbered from 7 to 9
9.7	Add section for script to update abs coefs after recal
9.9	Clarify B1U/GAC differences to B3 for HBT tables

TABLE OF CONTENTS

1	ISCCP – Getting Started	1
1.1	Porting and initial setup	1
1.1.1	Production system directory structure	1
1.1.2	Ancillary data locations	2
1.1.3	Input data locations	2
1.1.4	Working directories (wrkdirs)	3
1.1.5	Satellite positions (satpos).....	3
1.1.6	Satellite history (sathist).....	3
1.2	Building the system	3
1.2.1	3rd party software required.....	3
1.2.2	Set shell environment variables.....	4
1.2.3	Compile the Fortran code and install into prd/bin	4
1.2.4	Stage the input data	4
1.3	Running production	4
1.3.1	Schedule production	4
1.3.2	Start production	5
1.3.3	Restarting after an error.....	5
1.3.4	Production environment.....	5
1.3.5	Production cycle	5
1.3.6	Project dates (prjdat)	6
1.3.7	Timeslots (nomtim)	6
1.3.8	Re-calibration (RECAL)	7
1.3.9	Parallel processing	7
1.3.10	Product metadata	7
1.3.11	H-series product names.....	7
1.3.12	H-series filename conventions:.....	8
2	ISCCP Data Flow and Process Diagrams	9
2.1	Data flow diagram	9
2.2	Procedures for RE-processing	10
2.3	Procedures for NEW processing.....	11
3	Monitoring Production	12
3.1	Pre-production.....	12
3.1.1	SPC status reports (ALWAYS EXAMINE for EXTENDED PERIOD).....	12
3.1.2	QC Plots (ALWAYS EXAMINE).....	12
3.1.3	QC Results	12
3.1.4	CAL monitor plots (under development)	12
3.1.5	Ancillary Data monitor Plots (AS NECESSARY)	12
3.1.6	Additional NCDC checks	12

3.2	Production schedule (ALWAYS EXAMINE)	12
3.3	During production	13
3.3.1	Satellite hierarchy plots (<i>sathier</i>) (ALWAYS EXAMINE)	14
3.3.2	Preliminary cloud detection plots (<i>b4plot</i>) (ALWAYS EXAMINE).....	15
3.3.3	HGS stats (<i>dsstat</i>) (AS NEEDED for INVESTIGATIONS)	16
3.3.4	HGG stats (<i>d1stat</i>) (ALWAYS EXAMINE)	16
3.3.5	Console warnings from <i>prdrun</i> (ALWAYS EXAMINE)	18
3.3.6	Global log files (<i>wrkdir/log</i>) (AS NEEDED for INVESTIGATION).....	18
3.3.7	Satellite log files (<i>wrkdir/satpos/log</i>) (AS NEEDED for INVESTIGATION).....	18
3.4	Post-production - BASE PERIOD.....	19
3.4.1	HGM-D2 Comparison Map Plots and Summary Table (ALWAYS EXAMINE).....	19
3.4.2	HGG-D1 Comparison Histogram Plots (AS NEEDED for INVESTIGATION)	20
3.4.3	HXS-DX Comparison Histogram Plots (AS NEEDED for INVESTIGATION).....	20
3.5	Post-production – EXTENDED PERIOD.....	20
3.5.1	HGM-HGM Comparison Map Plots & Summary Table (ALWAYS EXAMINE).....	20
3.5.2	HGG-HGG Comparison Histogram Plots (AS NEEDED for INVESTIGATION)	20
3.5.3	HXS-HXS Comparison Histogram Plots (AS NEEDED for INVESTIGATION)	20
3.6	Isccpview.....	20
3.6.1	Building isccpview	21
3.6.2	Running isccpview	21
3.6.3	Opening data files.....	21
3.6.4	Viewing maps	21
3.6.5	Viewing cell contents	21
3.7	Troubleshooting	21
3.7.1	Error handling system.....	21
3.7.2	Warnings	22
3.7.3	Errors.....	22
3.7.4	Investigating	22
3.7.5	Re-starting.....	22
4	System architecture	24
4.1	Production data flow overview.....	24
4.2	Production cycle.....	25
4.2.1	Environment variables (<i>prdenv</i>).....	25
4.2.2	Scheduling GUI (<i>prdsched</i>)	25
4.2.3	Top-level run script (<i>prdrun</i>)	25
4.3	Production modes	27
4.3.1	Calendar Mode (normal production)	27
4.3.2	Sliding Mode (not used).....	27

4.4	Pre-processing steps	29
4.4.1	Ancillary data pre-processing (nnhirspak).....	29
4.4.2	Satellite hierarchy (sathier).....	29
4.4.3	Auxilliary data preparation (aux).....	29
4.5	Post-processing steps.....	29
4.6	Cloud detection and retrieval steps.....	30
4.6.1	Mapping (b4prod).....	31
4.6.2	Clear-sky composites (clrsky).....	35
4.6.3	Preliminary cloud thresholding (bxprod).....	36
4.6.4	Revised clear-sky and re-thresholding (cyprod).....	37
4.6.5	Cloud and surface property retrievals (dxprod).....	38
4.7	Gridded product steps.....	40
4.7.1	Gridding (dsprod).....	40
4.7.2	VIS-correction (viscor).....	40
4.7.3	Merging (d1prod).....	40
4.7.4	Calibration test.....	40
4.7.5	Filling (d1fill).....	41
4.7.6	Averaging (d2prod).....	44
4.8	Global pixel-level product (dxg, dxgfill).....	44
4.9	Code organization.....	45
4.9.1	prd/bin.....	45
4.9.2	prd/lib.....	46
4.9.3	prd/src.....	47
4.10	Data read/write modules.....	50
4.10.1	Equal area grid.....	50
4.10.2	Auxilliary data.....	50
4.10.3	Ancillary data.....	50
4.10.4	Satellite data.....	50
4.10.5	Intermediate step output.....	50
4.10.6	Products.....	51
4.11	Utilities.....	51
4.12	Tables (prd/tables).....	52
5	Satellite history sub-system	56
5.1	Overview.....	56
5.2	Satellite definition module (Sat.pm).....	56
5.2.1	Sample entry.....	58
5.2.2	Adding a new satellite.....	59
5.2.3	Changing the location of a satellite, or its SPC assignment.....	60
5.3	Satellite processing centers (Spc.pm).....	60
5.3.1	Adding a new SPC.....	60

- 5.4 Satellite positions (Satpos.pm).....60
 - 5.4.1 Adding a new satpos61
- 5.5 Satellite history events (Satevents.txt)61
 - 5.5.1 SUBSATLON keyword.....62
 - 5.5.2 SUBSTITUTE_SATKEY keyword.....62
 - 5.5.3 Entry format.....62
- 5.6 Sathist lookup function (Sathist.pm).....62
- 5.7 PTRANS tables63
- 5.8 REFGEOTABLES64
- 5.9 Radius.....64
- 5.10 Initial calibration setup (nominal calibration, HBT V0).....65
- 6 B1U/GAC Quality Control (B1QC) 66**
- 6.1 Statistics computed66
- 6.2 Running B1QC67
- 6.3 Plotting results (qcresults)67
- 6.4 Plot interpretation68
- 6.5 Visual inspections70
- 7 Initial Calibration Setup 74**
- 7.1 Overview74
 - 7.1.1 Nominal calibration (NOM)74
 - 7.1.2 Normalized calibration (NORM).....74
 - 7.1.3 Absolute calibration (ABS).....74
- 7.2 New satellite setup (NOM)74
 - 7.2.1 Response functions74
 - 7.2.2 Solar constant (SOLCON)75
 - 7.2.3 Polar orbiter nominal calibration (SATINFO)75
 - 7.2.4 Thermal channel tables (SATTABLE).....75
- 7.3 Initial coefficient files (NORM, ABS).....78
 - 7.3.1 NORM file format.....78
 - 7.3.2 ABS file format.....78
- 7.4 Initial V0 HBT files.....79
 - 7.4.1 Geostationary (b1uhbt).....79
 - 7.4.2 Polar orbiter (gachbt).....80
- 8 Monitoring Calibration 81**
- 8.1 Polar orbiter (MONITOR)81
- 8.2 Geostationary (NORM)81

8.3 Updating HBT with new coefficients.....81

9 RECAL Procedure 82

9.1 Determining whsether RECAL is necessary82

9.2 Computing new IR and VIS offsets82

9.3 Investigating warnings about opposite signs.....82

9.4 Generating a new version of HBT tables.....83

9.5 Re-running the satellites83

9.6 Finishing production83

9.7 Updating the ABS coefficient files84

9.8 Sample RECAL scenario84

9.9 HBT format.....86

1 ISCCP – Getting Started

This section contains the basic instructions for building and running the ISCCP H-series production system. It is divided into three sections: porting and initial setup, building the system, running the system. See also “ISCCP_dataflow_diagram.pdf” and the various README files.

1.1 Porting and initial setup

Ftp the distribution tar file to the new system and unpack it in desired location (referred to as ISCCP_ROOT below). This creates the production system directory structure.

1.1.1 Production system directory structure

```
ISCCP_ROOT/
  isccpview/
    IDL application for viewing products
  doc/
    Documents and flowcharts
  calibration/
    D-coefficients/
      norm/      Final NORM coefficients for D-data (NORM-D)
      abs/      Final ABS coefficients for D-data (ABS-D)
    H-coefficients/
      norm/      Current NORM coefficients for H-data (NORM-H = NORM-D)
      abs/      Current ABS coefficients for H-data (ABS-H)
      D_abs_to_H_abs_coefs/ Coefficients applied to ABS-D to get ABS-H
    nominal/     Nominal calibration info files for polar orbiters
    sat_response_functions/ Tables of wavelength,response for each sat
  comparison_plots/
    hgm/        IDL plot package for HGM-D2 or HGM-HGM
    hgg/        IDL plot package for HGG-D1 or HGG-HGG
    hxs/        IDL plot package for HXS-DX or HXS-HXS
  prd/
    bin/
      Perl scripts to run the system
    lib/
      Perl modules and libraries, satellite history, product metadata
    src/
      Fortran source code
    tables/
      Static look-up tables used in production, srftyp and topo datasets, equal grid info
    ancil/
      aerosol -> Aerosol data (monthly files)
      nnhirs -> NNHIRS atmospheric data (3-hour files in yearly sub-dirs)
      ozone -> Ozone data (daily files in yearly sub-dirs)
      snowice -> Snowice data (daily files in yearly sub-dirs)
  data/
    blu -> B1U input data, in satpos sub-dirs
    gac -> GAC input data, in satpos sub-dirs
```

```

b3 ->  B3 input data, in satpos sub-dirs
b3qc -> B3QC data (only needed for B3 input data)
bt ->  HBT calibration data, in satpos sub-dirs
wrkdirs/
Working directories, one per production run.
Wrkdirs contain sub-dirs for each satellite position.
Sub-dirs created automatically when production is scheduled.

yyyy_mm/ (wrkdir name can be anything you want, empty at first)
  gms/ (satellite sub-dir)
    yyyy_mm.hist (hist files are created
                  automatically by sathist)
  b4/  b4 step output
  clrsky/ clrsky step output
  bb/  stats for b4plot
  bx/  bx step output
  cy/  cy step output
  dx/  hxs products
  ds/  hgs products
  log/ log files from production)
  goe/ (same as above for each satellite position)
  gow/
  ins/
  met/
  noa/
  nom/
  aux/ (auxiliary data, generated by auxgeo/auxpol)
  tbnn/ (NNHIRS brightness data, generated by nnhirspak)
  sathier/ (satellite hierarchy, generated by sathier)
  d1/  hgg products
  d2/  hgh/hgm products
  dxg/ hxg products
  log/ log files

```

As production runs, sub-directories will be created for each processing step, within each satellite position sub-directory in the working directory that was created within prd/wrkdirs. To add new satellite positions see Section 5.4.

1.1.2 Ancillary data locations

The distribution tar file contains ancillary data for July 2007 (AEROSOL, NNHIRS, OZONE, SNOWICE) in the directory prd/ancil. For other time periods, make links here to point to the staged data. All ancillary data products are available for 1979-2013. Other time periods will need to have ancillary data produced. The ancillary data content and production process is documented separately in the ISCCP Ancillary Production Guide.

1.1.3 Input data locations

The distribution tar file contains HBT calibration tables for July 2007 in the directory prd/data/bt. For B1U/GAC processing, create links for: B1U and GAC data (system can also run from B3 with links for B3 and B3QC, where B3QC points to the B3 QCRESULTS files which

are available through 2009). See the README.txt files in the prd/data/b1u and prd/data/gac directories for details.

1.1.4 Working directories (wrkdirs)

To schedule production, create a directory within the prd/wrkdirs directory and cd into it, or create link to directories on other locations. Copy prd/bin/prdenv.template to prdenv in the directory you have created, edit as needed and source it to set the environment variables. Then run prdsched to schedule a production run. All sub-directories will be automatically created as production runs. All output will be contained within the wrkdir you have created. Multiple wrkdirs can be set up at the same time. See “Running Production” section below for more details.

1.1.5 Satellite positions (satpos)

ISCCP divides the globe into 7 satellite positions. These are:

```
gms : Japan sector geosat
goe : Goes-east sector geosat
gow : Goes-west sector geosat
ins : Indian ocean sector geosat
met : Meteosat sector geosat
noa : NOAA afternoon orbiter
nom : NOAA morning orbiter
```

When setting up links to staged data, use the 3-letter acronym for each position. To add new satellite positions see Section 5.4

1.1.6 Satellite history (sathist)

Various satellites occupy each sector at any given time, and are processed by various centers, as specified in prd/lib/Sat* modules.

Sat.pm – individual satellite descriptions such as names, number of channels, etc.

Satpos.pm – satellite position definitions

Sathist.pm – interface for production system to look-up satellite info for given date

Sathist.events – table of satellite events such as start/stop dates, moves, problems, etc.

Spc.pm – sector processing center definitions

When adding a satellite to the system, these files will need to be modified (see Section 5).

1.2 Building the system

1.2.1 3rd party software required

1. Tk.pm perl module, available from www.cpan.org
2. Netcdf 4, which requires also HDF 5
3. NCO - Netcdf operators, available from nco.sourceforge.net

4. IDL for plots

1.2.2 Set shell environment variables

1. Go to the prd/bin directory
2. Copy prdenv.template to prdenv
3. Edit prdenv to set ISCCP_ROOT to the path to the "prd" directory. Default production mode is calendar mode (one calendar month at a time) for 10km data input, 1.00 degree grid output.
4. Set IDL_CMD to the appropriate command for your system to run the IDL plots
5. Source it, eg. ". ./prdenv" (make sure you do this for every shell window you open and keep a copy in each wrkdir so you can have different setups for each)

1.2.3 Compile the Fortran code and install into prd/bin

1. Go to the prd/src directory
2. Copy Makefile.template to Makefile
3. "make"
4. "make install" (just copies executables into the prd/bin directory)
5. To compile with debugging compiler options "make -B DEBUG=yes; make install"

1.2.4 Stage the input data

1. Stage the BIU, GAC, AEROSOL, NNHIRS, OZONE, SNOWICE and HBT data (TOPO and SURFACETYPE are already provided in the prd/tables directory). Note that some steps require +/- some number of days around the dates being processed to be staged. See sections 4.6.1 (b4prod), 4.7.5 (d1fill) and 4.8 (dxgfill) for staging requirements for those steps.
2. Create links in prd/data and prd/ancil to point to the staged data

1.3 Running production

1.3.1 Schedule production

1. Create a directory or link within prd/wrkdirs and cd into it. Copy prd/bin/prdenv and edit as needed, and source it to set the environment variables.
2. Run prdsched: "prdsched yyyybeg mmbeg ddbeg yyyyend mmend ddend" with the dates you want to schedule for running (requires Tk.pm perl module, available from www.cpan.org). This is the gui for scheduling a production run. When you make selections and "accept", it creates the text file "tasklist". To skip the gui and just create a full tasklist, append the arg -nogui to the prdsched command line, eg: "prdsched yyyybeg mmbeg ddbeg yyyyend mmend ddend -nogui".

1.3.2 Start production

1. After running prdsched you are ready to start the production run.
2. "nohup prdrun &> prdrun.out &" or "prdrun | tee prdrun.out"
3. Production will run until an error is encountered or all tasks are done
4. Check caltest results to see if re-cal is needed
5. Check logs for diagnostic output
6. Check b4plot, dsstat, d1stat (IDL plots)
7. View product maps with isccpview IDL application (see README in isccpview dir)

1.3.3 Restarting after an error

1. Check logs to find the error and fix it
2. Run prdsched (no args) to adjust the schedule if necessary. By default, the run will pick up where it left off.
3. nohup prdrun &> prdrun.out &

1.3.4 Production environment

Production configuration is controlled via environment variables that are set via the prdenv script, that are unique to each wrkdir. These variables define the locations of inputs and set various switches for the code. Some of the more useful switches control whether to keep or remove intermediate output files. See section 4.2.1 for environment variable descriptions.

1.3.5 Production cycle

The prdrun script runs tasks according to the tasklist file which is a text file that can be edited with prdsched, or by hand. Each step is a separate Fortran code, which is run via a Perl script. See also "ISCCP_dataflow_diagram.pdf" and the README file in prd/src. The steps are:

Pre-processing:

nnhirspak – set up brightness temperature profiles

sathier – set up satellite hierarchy based on actual satellite locations

Then the main algorithm runs separately for each satellite, looping over timeslots:

aux – setup ancil data for the satellite

b4prod – calibration, mapping, space/time tests, runs all timeslots

clrsky – construct clear-sky maps

bxprod – threshold

cyprod – revise clear-sky maps, re-threshold

dxprod – retrievals (HXS product)

dsprod – gridding (HGS product)

Then, for all satellites together:

d1prod – merging (HGG product, un-filled)
d1fill – fill empty cells (HGG product, filled)
d2prod – monthly means (HGH/HGM products)

Post-processing:

During merging, statistics from overlapping satellites are collected and analyzed to see if re-calibration is required. Steps involved in this are:

d2calcor – computes calibration correction from overlap stats
caltest – decide if re-cal is needed (manually create a new version of HBT and re-run, see Section 9)

After all other steps are finished, the HXS is merged into HXG pixel-global product and filled using these steps:

dxgprod – create HXG product, unfilled
dxgfill – fill HXG product

Plots/statistics:

After certain steps, some plots/tables are automatically created for monitoring results.

b4plot – after bxprod, plots radiances
dsd1stat – after dsprod for each sat, and after d1prod for global, generates statistics tables/plots

1.3.6 Project dates (prjdat)

The production system uses a sequential project date throughout (prjdat). Project date 1 is July 1, 1983, and continues sequentially to the present day. Functions are provided to convert calendar dates to project dates and vice versa. For processing data prior to July 1, 1983, negative project dates are used. Project date 0 is skipped. Project date -1 is June 30, 1983, project date -2 is June 29, 1983, and continue to count backwards in time.

1.3.7 Timeslots (nomtim)

ISCCP products are produced at eight nominal times per day. Each nominal time (00,03,06,09,12,15,18,21 GMT) represents a 3-hour time window. This window is referred to as a timeslot, called “nomtim” for processing. The actual data time may be anywhere within this 3-hour window of +/- 1.5 hours around the nominal time. Note that nominal time 00 may include data from 22:30 of the previous day.

For geostationary satellites there is one image file available per 3-hour time window, so the algorithm is run separately for each 3-hour nominal time (8 times per day). Thus, for geostationary data, nomtim ranges from 1 to 8, corresponding to the 3-hour windows around the nominal times of 00,03,06,09,12,15,18,21 GMT.

For polar orbiters there may be multiple orbit files available that contain data within a 3-hour window. The first processing step inputs the multiple orbit files for each day, discards duplicates

(orbits whose start/end times are contained within other orbits are skipped in Filenames.pm when generating list of input files for either b4prod or b1uqc) and sorts the orbits according to individual scan line times into 18 separate processing times. Thus, for polar orbiters, nomtim ranges from 1 to 18 for each day:

T 1-8 = south pole scans for each 3-hour timeslot (1=00, 2=03 ... 8=21 GMT)

T 9-16 = north pole scans for each 3-hour timeslot (9=00, 10=-3 ... 16= 21 GMT)

T 17 = mid-latitude scans from ascending portion of orbits over the whole 24-hour period

T 18 = mid-latitude scans from descending portion of orbits for the whole 24-hour period

1.3.8 Re-calibration (RECAL)

At the end of the run, the results from the analysis of the overlapping satellite statistics (caltest) will indicate whether re-calibration is necessary. These results are in the log file caltest.pl.out.nn. If the results indicate that RECAL is required for any satellite, then a new version of HBT files needs to be created with the indicated adjustments, and production re-run from the beginning using the new version of HBT files. This process iterates until no further adjustments are required. For instructions on running the RECAL process see Section 9. HBT Version 0 is provided for July 1983 through 2009. For other time periods initial HBT files will need to be created through a separate process.

1.3.9 Parallel processing

Because each run is self-contained in a wrkdir, multiple runs can be initiated at the same time, providing a form of parallel processing. To parallel process within a single run, the series of tasks for each satellite can be run in parallel up to the gridding step, and then all must merge together. An alternate prdrun script (parallel_prdrun) is provided in prd/bin which does this using the Perl ForkManager module (available from www.cpan.org).

1.3.10 Product metadata

The NetCDF product files are generated from CDL to create a stub file with metadata (using ncgen). The Fortran production code then opens this stub file and fills in the variables. The metadata can be altered by editing the CDL files in prd/lib/formats. Note that the HGS CDL variable definitions are in the form of comments (all lines preceeded by //). That is because the dsprod.f90 parses it instead of ncgen, to create the NetCDF variables and attributes that are specified, since it is not known ahead of time how many cells are to be written.

1.3.11 H-series product names

Much of the code still uses D-series prefixes for program names and variables. This is the correspondence:

D-series	vs	H-series
dx		hxs, hxg
ds		hgs
d1		hgg
d2		hgh, hgm

H-series names are three letters:

1. H
2. X = pixel or G = gridded
3. S = single satellite or G = global, H = hourly-monthly, M = monthly

1.3.12 H-series filename conventions:

Intermediate step output for mapping and clear-sky

```
B4.sssn.ddddd.Ttt.dat
BB.sssn.ddddd.Ttt.dat
CLRSKY.sssn.ddddd.Ttt.dat
```

where:

```
sssn = sat n
dddd = prjdat
tt = timeslot 1-18
```

Intermediate step output for thresholding

```
BX1ssnt.yymmddgg
CY1ssnt.yymmddgg
```

where:

```
sssn = sat n
t = type 0,S,N,A,D
yymmddgg = year month day gmt
```

Single-satellite products

```
ISCCP.HXS.v.sssn.yyyy.mm.dd.gg00.ccc.nnKM.CSppppppppppp (geostationary)
ISCCP.HXS.v.sssn.yyyy.mm.dd.gg00.ccc.nnKM.CSppppppppppp.t (polar orbiter)
ISCCP.HGS.v.sssn.yyyy.mm.dd.gg00.ccc.nnKM.CSppppppppppp.EQgggg.nc
```

where:

```
v = version number
sssn = sat n or GLOBAL
yyyy.mm.dd.gg00 = year month day gmt
ccc = processing center or GPC
nnKM = 10km or 30km resolution
CSppppppppppp = prjbeg prjend that was used for clrsky
t = S,N,A,B,C,D,E,F for polar orbiter HXS only
gggg = grid size (1.00, 2.50)
```

Global products

```
ISCCP.HGG.0.GLOBAL.yyyy.mm.dd.gg00.GPC.nnKM.CScc.EQgggg.nc
ISCCP.HGH.0.GLOBAL.yyyy.mm.99.gg99.GPC.nnKM.CScc.EQgggg.nc
ISCCP.HGM.0.GLOBAL.yyyy.mm.99.9999.GPC.nnKM.CScc.EQgggg.nc
```

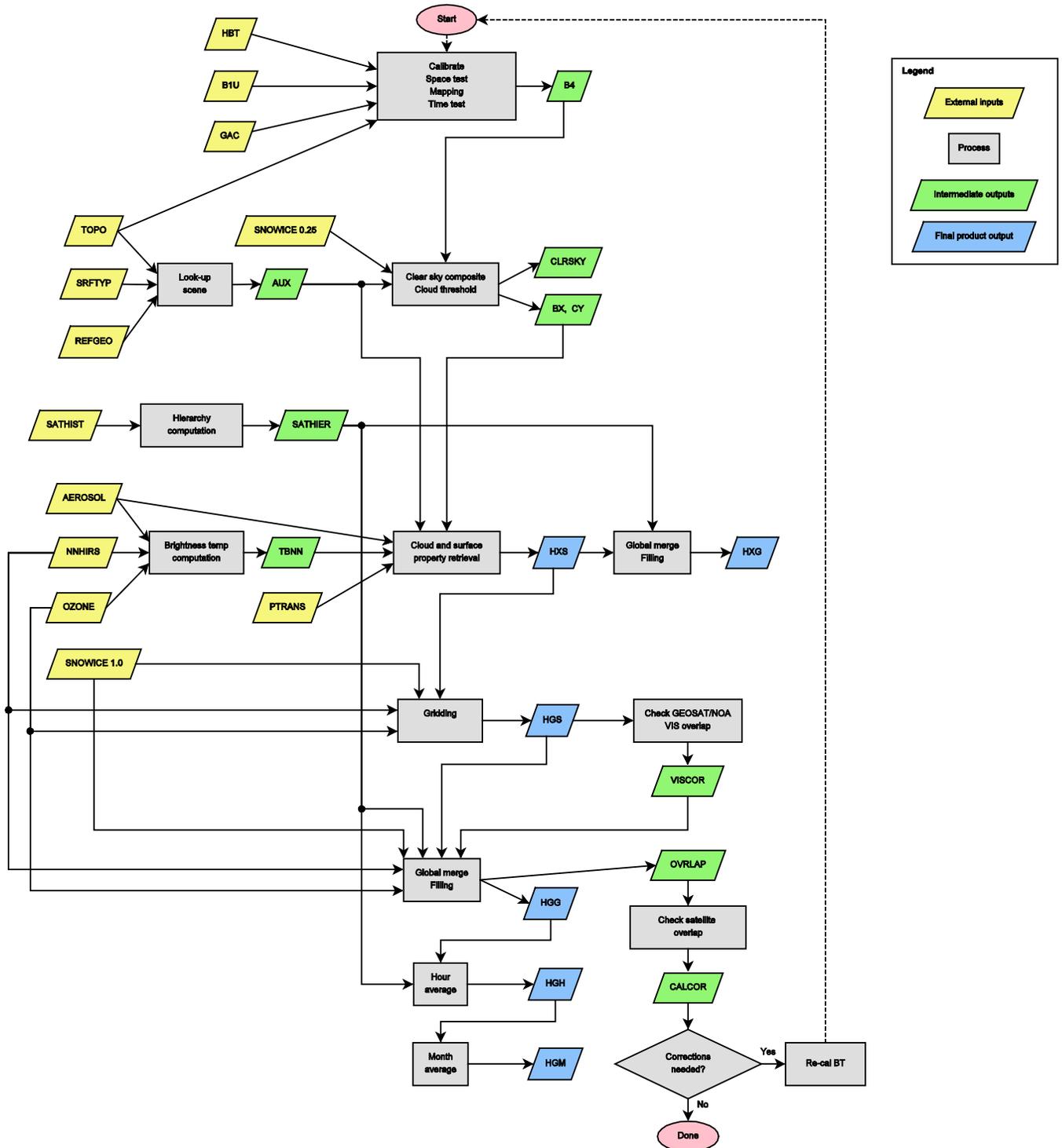
where:

```
nn = 10 or 30 (input resolution)
cc = 12 or 17 or 00 (clear-sky mode 12-sliding, 17-sliding, 00-
calendar)
gggg = 1.00 or 2.50 (grid size)
```

2 ISCCP Data Flow and Process Diagrams

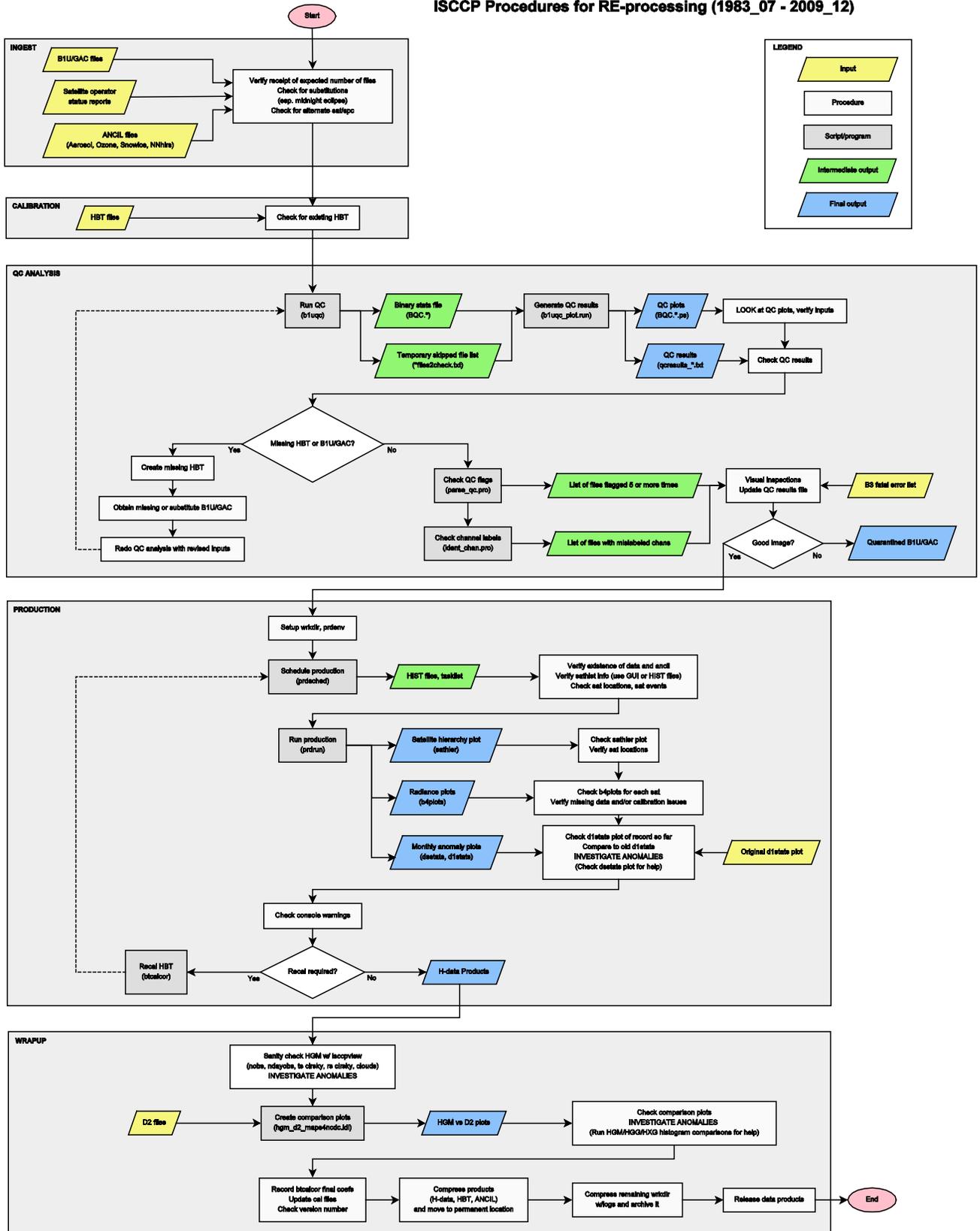
2.1 Data flow diagram

ISCCP H-Series Data Flow

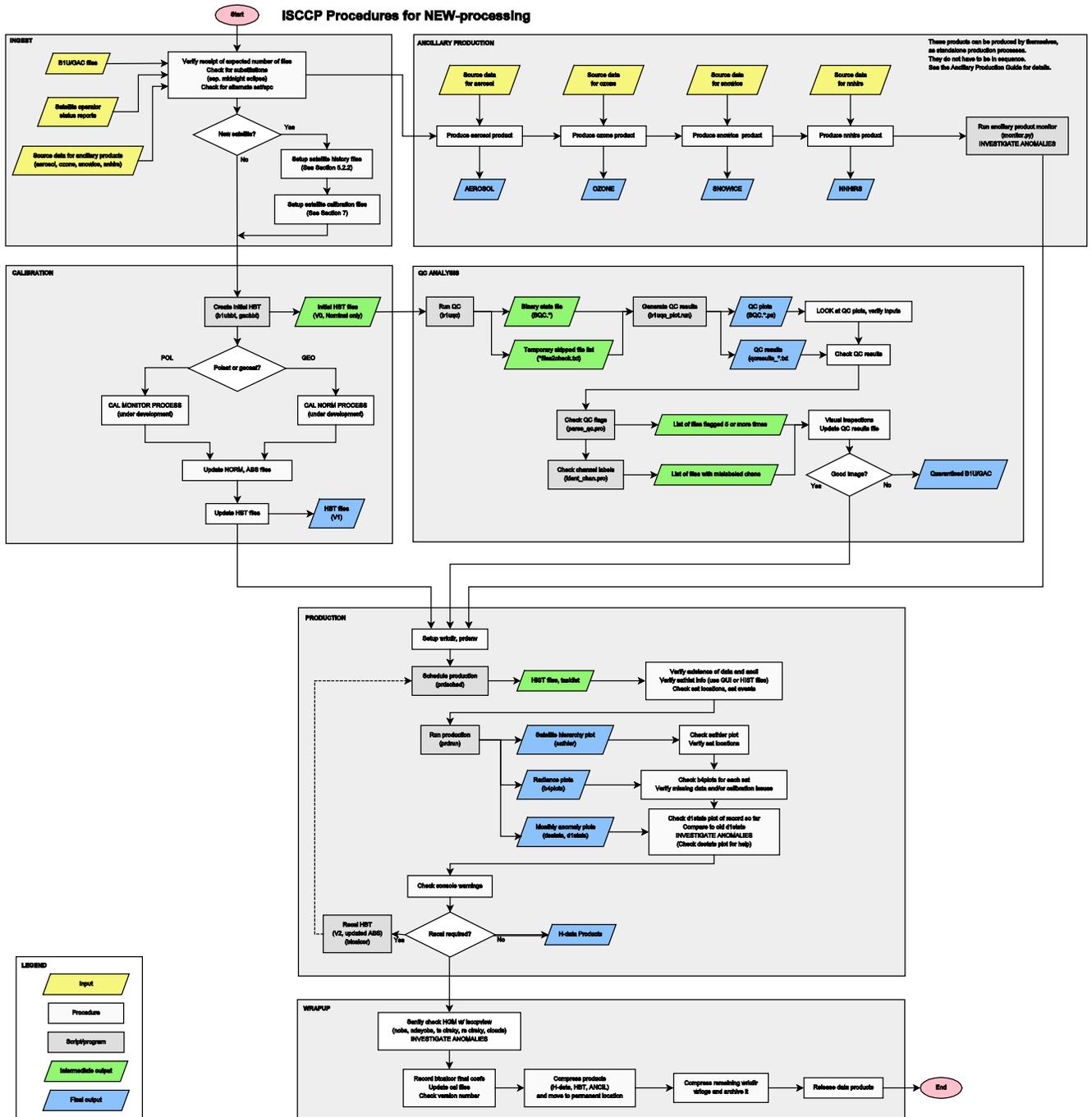


2.2 Procedures for RE-processing

ISCCP Procedures for RE-processing (1983_07 - 2009_12)



2.3 Procedures for NEW processing



3 Monitoring Production

This section contains basic guidelines for monitoring production and describes the various logs, statistics and plots that are generated to make sure things are running properly. See also the procedure flowcharts in Sections 2.2 and 2.3.

3.1 Pre-production

3.1.1 SPC status reports (**ALWAYS EXAMINE** for EXTENDED PERIOD)

These should be checked to determine the number of image files delivered, whether there are any substitutions for the nominal time slot images, especially during eclipse seasons, and whether there are any alternate satellites used to provide data. (Also, note should be made of whether images from a particular satellite were obtained from an alternate SPC.)

3.1.2 QC Plots (**ALWAYS EXAMINE**)

The QC plots for all GAC and B1U inputs should always be examined. This means looking at them to check for consistency and to identify outliers that may indicate problems needing to be addressed before production can begin. See Section 6 for description of plot contents and interpretation.

3.1.3 QC Results

The QC results text file lists images flagged as questionable by the B1QC program, as well as images that were skipped due to missing HBT or inability to be read. See Section 6 for description of file contents.

3.1.4 CAL monitor plots (under development)

The calibration of all satellites is monitored over time to derive updates to the normalized and absolute calibration coefficients. The various monitor processes produce plots, see Section 8 for description of plot contents (under development).

3.1.5 Ancillary Data monitor Plots (**AS NECESSARY**)

The pre-processing codes for each of the time-varying Ancillary Datasets (AEROSOL, OZONE, NNHIRS, SNOWICE) produce an up-to-date record of the global monthly mean values of the main parameters since 1980. This record may be plotted as needed using the ancillary data monitor program. These values should not exhibit any sudden changes over time. See the ISCCP Ancillary Production Guide for more information about running the monitor.

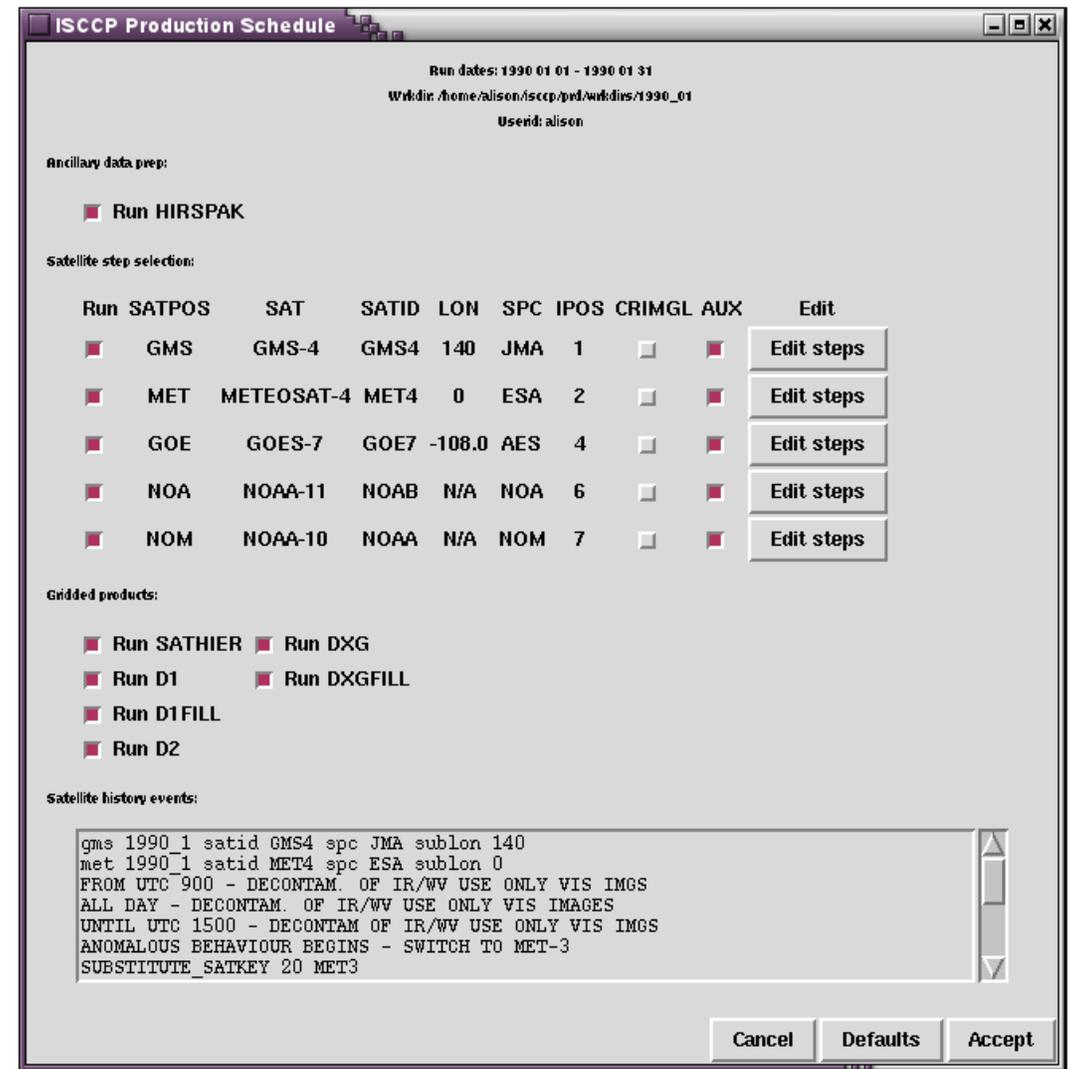
3.1.6 Additional NCDC checks

In addition to the ISCCP pre-processing checks, a review of the NCDC image quarantine list and channel ID check results should be performed before starting production.

3.2 Production schedule (**ALWAYS EXAMINE**)

When *prdsched* is used to schedule production, the satellite history (see Section 5) is checked for information about the actual satellite configuration for the month being processed. This information is presented on the GUI for review and confirmation. This information is simultaneously written to text files (HIST files) within each satellite sub-directory (*wrkdir/satpos/yyyy_mm.hist*) which can also be viewed manually. At any time after initial scheduling, re-running *prdsched* without any args will show the configuration information. This information should always be examined to check the satellite configuration and events, using either the GUI or by looking at the HIST files.

Fig 1: Sample GUI screens



3.3 During production

3.3.1 Satellite hierarchy plots (*sathier*) (**ALWAYS EXAMINE**)

The *sathier* step generates the satellite hierarchy. During production this happens twice, once for the 1.0 degree HGG merge, and a second time for the 10km DXG merge. The hierarchy results are written to a binary file in the `wkdir/sathier` directory. A graphical representation of the hierarchy is plotted as a PNG image also in this directory. This should be looked at to make sure the satellites are being placed in the proper positions. This is especially important during times of moving satellites. The plot program is `prd/bin/idl/sathier_plot.pro`. If IDL is available to production the plot output will be in the `wkdir/sathier` directory.

Figure 2 shows an example of the satellite hierarchy plot for the month of July 2007. The four levels of satellite preference are as follows. Level 1 (top preference) is a geostationary satellite in each longitude sector equatorward of 55° N and S with the smallest satellite view angle. Poleward of 55° the afternoon polar orbiter is the top preference. Level 2 (if a Level 1 image is not available) is a geostationary satellite in an adjacent longitude sector equatorward of 55° N and S that has a view of that particular location. Poleward of 55° the morning polar orbiter is the second preference. Level 3 (if no Level 1 or Level 2 image is available) is either the afternoon or morning polar orbiter equatorward of 55° N and S and is any available geostationary view poleward of 55° . Level 4 (if no image at a higher level is available) is the morning polar orbiter equatorward of 55° N and S.

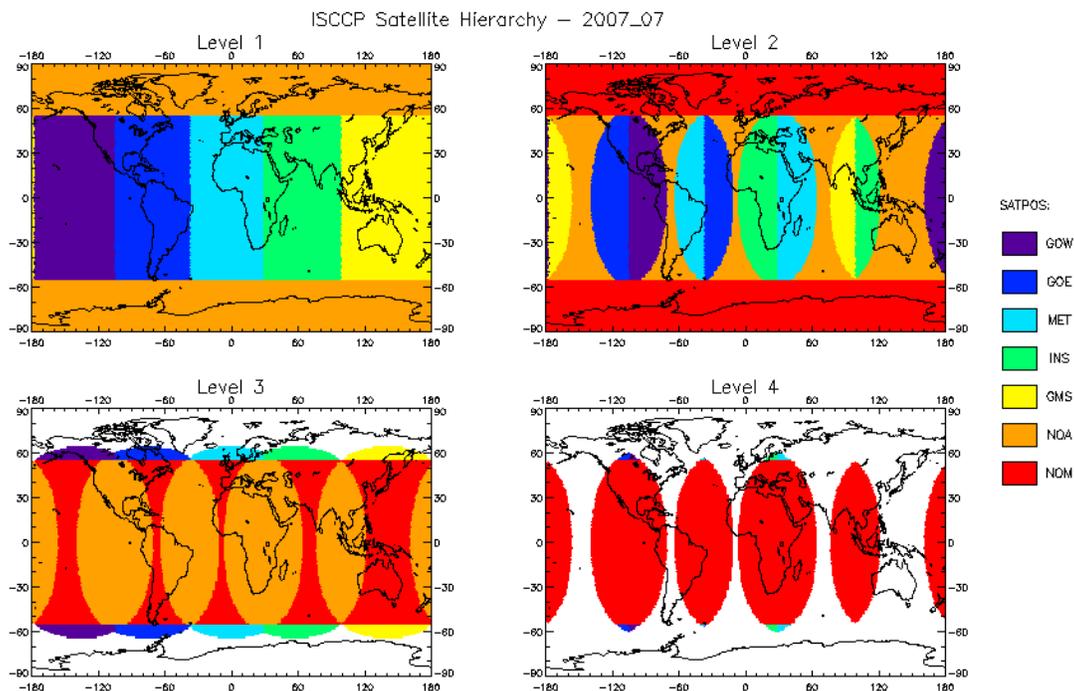


Fig 2: July 2007 satellite hierarchy plot for the 1 degree grid.

3.3.2 Preliminary cloud detection plots (*b4plot*) (ALWAYS EXAMINE)

The initial clear-sky determination and cloud thresholding is done in the *bxprod* step, and later refined in the *cyprod* step. As a diagnostic, the *bxprod* step outputs both *bx* data files and *bb* summary statistics. The *bb* files are little text files that contain mean cloud amounts and clear-sky radiances at each timeslot each day. These are input to the *b4plot* programs which are automatically run after *bxprod* finishes. The plot programs are *prd/bin/idl/b4plot_geo.pro* and *prd/bin/idl/b4plot_pol.pro*. If IDL is available to production these plots will be in each *wrkdir/satpos/b4plot* directory. These plots should be looked at to monitor the radiances and to easily identify missing data.

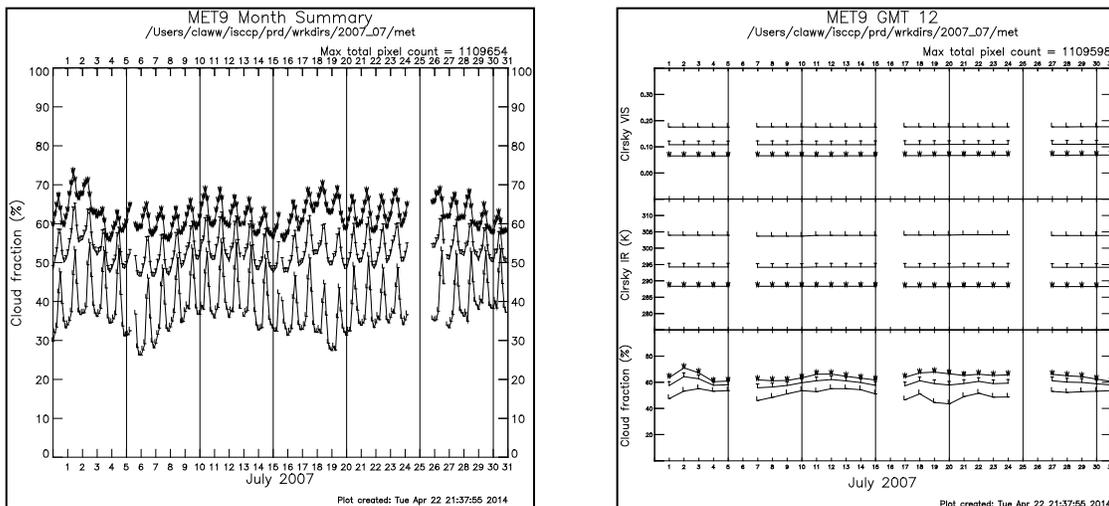


Fig 3: *B4plots* for MET July 2007. On the left is the summary page for the whole month. On the right is the page for timeslot 5 (GMT 12). There will be a page for each timeslot. For the polar orbiters, the plots will be divided into North, South, Ascending and Descending pages.

3.3.2.1 Geostationary satellite *B4plot* interpretation

Page 1: Month Summary – The plot shows the fraction (%) of image pixels present (unmarked line near top) with respect to the maximum count printed upper right and the image cloud fraction (%), divided into water (W), land (L) and total (T) for each image in the month. Although there can be notable diurnal variation in cloud amount, especially over land, there should not be large deviations of image cloud amounts, except possibly but when the image is truncated as shown by a large decrease of the fractions of pixels present.

Pages 2-9: These pages of plots, one for each GMT (time) slot, show the image average clear sky VIS (0-1) and clear sky IR (Kelvin) radiances and the cloud fraction (%), divided into water (W), land (L) and total (T) for each day. The bottom plot also shows pixel fraction (unmarked line, relative to maximum value printed). Given that solar illumination occurs only at certain GMTs, different but characteristic for each geostationary position, some clear sky VIS plots should show very low values all month (no sunlight) and higher values during daylight times. Clear sky IR should be very stable over the month, especially for water areas. Sudden decreases in pixel fraction can be associated with altered clear sky radiances.

3.3.2.2 Polar orbiter B4plot interpretation

Page 1-4: Unlike the geostationary satellite data that provide whole-Earth views eight times per day, the polar orbiting data combines nearly 14 orbits per day into a daily image. For these plots, the Earth is divided into three geographic sectors – north and south polar (55° to the pole) and lower (–55° to 55°) latitudes; the statistics for the latter sector are presented for ascending and descending orbit segments (ascending/descending segments are generally daytime/nighttime for afternoon orbiters and nighttime/daytime for morning orbiters). The plots show the average clear sky VIS (0-1) and clear sky IR (Kelvin) radiances and the cloud fraction (%), divided into water (W), land (L) and total (T) for each day. The bottom plot also shows pixel fraction (unmarked line, relative to maximum value printed). Clear sky VIS and IR values should be very stable over the month, especially for water areas. Sudden decreases in pixel fraction can be associated with altered clear sky radiances.

3.3.3 HGS stats (dsstat) (AS NEEDED for INVESTIGATIONS)

After the *dsprod* step finishes producing the HGS files, the *dsd1stat* step is run to accumulate statistics over the month from the files that were produced. It prints a table of means for selected variables broken down by day/night, land/water and hemisphere/global. This table is written to a text file in the *wrkdir/satpos/ds* directory along with the HGS files. This table is also copied to a permanent location in the *prd/dsstat* directory so that it can later be plotted over time. The filename is *dsstat.satpos.yyyy_mm*, it is a text file.

As the time record builds, these statistics are plotted to show the anomalies in the means of selected variables over time. The plot program is *prd/bin/idl/dstat_plot.pro*. If IDL is available to production, the plot will be created as a postscript file in each *wrkdir/satpos* directory. The filename is *dsstat.satpos.yyyy_mm-yyyy_mm.ps* which indicates the date range used as a basis for the anomalies. This will vary as the production time record increases.

These plots are similar to the D1stats plots in content, described in Section 3.3.4 below, but the DSstats show the statistics separately by satellite. If an anomaly is found in the D1stats plots, the DSstats plots should be examined to determine if the cause is associated with a single satellite. Variations in these plots may be somewhat larger than in the D1stats plots.

3.3.4 HGG stats (d1stat) (ALWAYS EXAMINE)

After the *d1prod* step finishes producing the HGG files (un-filled), the *dsd1stat* step is run to accumulate statistics over the month from the files that were produced. This is done before the *d1fill* step so these statistics are for un-filled HGG data. It prints a table of means for selected variables broken down by day/night, land/water and hemisphere/global. This table is written to a text file in the *wrkdir/d1* directory along with the HGG files. This table is also copied to a permanent location in the *prd/d1stat* directory so that it can later be plotted over time. The filename is *d1stat.yyyy_mm*, it is a text file. For investigation of filling issues, after the *d1fill* steps are done, statistics on number of cells filled can be found in the step logs, or the various *fill_type* codes can be surveyed in the filled HGG files.

As the time record builds, these statistics are plotted to show the anomalies in the global means over time for each variable. The plot program is *prd/bin/idl/dstat_plot.pro*. If IDL is available to production, the plot will be created as a postscript file in the main *wrkdir* directory. The filename

is d1stat.yyyy_mm-yyyy_mm.ps which indicates the date range used a basis for the anomalies. This will vary as the production time record increases.

These plots summarize the (growing) time record by displaying global monthly mean values of various quantities as anomalies from the record average (these averages change as more data are added to the record and are printed at the top of the plot, along with the standard deviation of the anomalies). The title notation indicates whether the averages are over daytime only (D) or over day and night (D, D.N). Most physical quantities are averaged in coded “count” values as is noted; otherwise quantities are averaged in physical units. **In general all plots should exhibit small and smooth variations; sudden and/or large variations are to be investigated.**

Page 1: The first three plots show the average number of satellite pixels per map grid cell, the number of map grid cells that have original content and the cloud fraction (%). The first two counts will vary if data are missing; large variations should be verified. Cloud fraction should not vary by large amounts (more than a couple of percent unless a lot of data are missing); large variation should be investigated. The next three plots show the frequency of occurrence (0-1) of cloud fractions in three ranges, 0-10%, 40-60% and 90-100%.

Pages 1-2: The last plot on Page 1 and the first six plots on Page 2 show the amounts of various cloud types as fraction (%). The first three plots show all low-level clouds (as determined during daytime by the sum of cumulus, CU, stratocumulus, SC, and stratus, ST), all middle-level (daytime sum of altocumulus, AC, altostratus, AS, and nimbostratus, NS), all high-level clouds (daytime sum of cirrus, CI, cirrostratus, CS, and cumulonimbus, CB). The next four plots show amounts for cirrus, cumulonimbus, cumulus and stratus. The variations over the record of these cloud types amount should be relatively small and smooth; sudden and/or large variations are to be investigated. The last four types are especially useful for diagnosing spurious changes caused by calibration problems as they represent the clouds associated with smallest (cirrus, cumulus) and largest (cumulonimbus, status) VIS radiances and with the smallest (cirrus, cumulonimbus) and largest (cumulus, stratus) IR radiances. Coordinated variation patterns can indicate which radiance calibration might be causing a problem: cirrus-cumulus and cumulonimbus-status suggests VIS related changes, whereas cirrus-cumulonimbus and cumulus-stratus suggests IR related changes.

Pages 2-3: The last plot on Page 2 and the first four plots on Page 3 display the main cloud and surface properties retrieved by the ISCCP analysis: cloud top pressure and temperature (from combined VIS and IR), cloud optical thickness (mainly from VIS), surface reflectance (from VIS) and surface (skin) temperature (from IR). These quantities should exhibit small and smooth variations; sudden changes may also point to calibration issues when coordinated: cloud top and surface temperature indicating IR related changes and cloud optical thickness and surface reflectance indicating VIS related changes.

Pages 3-4: The last three plots on Page 3 and the first four plots on Page 4 display statistics that come from the NNHIRS Ancillary product used in the ISCCP analysis: tropopause temperature, temperature at 380, 500 and 900 mb, tropopause pressure, surface pressure and near-surface relative humidity (RH). The next two plots come from the OZONE and SNOWICE Ancillary product. Time variations should be small and smooth; sudden changes should be investigated.

Pages 4-5: The last plot on Page 4 and the first three plots on Page 5 also come from the NNHIRS Ancillary product: near-surface air temperature, relative humidity (RH) at 380, 500 and 900 mb. The last four plots on Page 5 show the average cloud amounts as determined by IR-only tests in the analysis: low-level, middle-level and high-level amounts together with total cloud amount. Time variations should be small and smooth; sudden changes should be investigated.

Page 6: The first plot on Page 6 shows the cloud water path. Time variations should be small and smooth; sudden changes should be investigated.

Pages 6-11: The remaining plots illustrate the global monthly distributions of various quantities by showing the 5th, 50th and 95th percentile values, which may be helpful for diagnosing a problem. Quantities shown are: cloud top pressure and temperature, cloud optical thickness, surface reflectance and temperature, atmospheric temperature at 380, 500 and 900 mb, ozone amount, relative humidity at 380, 500 and 900 mb and cloud water path. The variability at the different percentiles shows characteristic patterns for each quantity: (a) cloud top pressure and temperature, surface temperature, atmospheric temperatures at 500 and 900 mb and relative humidity at 900 mb show decreasing variation magnitude from 5th to 95th percentile, (b) cloud optical thickness, surface reflectance, ozone, relative humidity at 380 and 500 mb and cloud water path show increasing variation magnitude from 5th to 95th percentile and (c) atmospheric temperatures at 380 mb show similar variation magnitudes for all percentiles. Time variations should be small and smooth; sudden changes should be investigated.

3.3.5 Console warnings from *prdrun* (ALWAYS EXAMINE)

As the *prdrun* script processes the task list, a timestamp with a message is printed at the beginning and end of each step that is run. The output from each step goes into a log file, and *prdrun* will scan the step log file after the step has finished looking for the words “WARNING” or “ERROR”. Any messages that a step program has printed with those words in the line are echoed to the console to bring it to the operator’s attention. The console log is the first place to check for warnings and errors, and to monitor the time of execution.

3.3.6 Global log files (*wrkdir/log*) (AS NEEDED for INVESTIGATION)

Some of the steps are global in nature, and these run in the main working directory (*wrkdir*). The output from these steps will be found in *wrkdir/log*. These are the setup steps (*sathier*, *nnhirspak*) the global merge and monthly mean steps (*d1prod*, *d2prod*, *dxgprod*) and the calibration check results (*caltest*). Anything that these step programs write to stdout (Fortran unit 6) will be in the log.

3.3.7 Satellite log files (*wrkdir/satpos/log*) (AS NEEDED for INVESTIGATION)

Some of the steps apply to individual satellites, and these run in individual satellite sub-directories of the *wrkdir*. The output from these steps will be found in *wrkdir/satpos/log*. These are the mapping steps (*aux*, *b4prod*) the cloud detection algorithm (*clrsky*, *bxprod*, *cyprod*) the retrievals (*dxprod*) and the gridding (*dsprod*). Anything that the step program prints to stdout (Fortran unit 6) will be in the log. The log file names for any steps that run in individual time slots are prefaced by Tn where n is the timeslot number (1-8 geo, 1-18 polar).

3.4 Post-production - BASE PERIOD

3.4.1 HGM-D2 Comparison Map Plots and Summary Table (ALWAYS EXAMINE)

This IDL plot package is located in comparison_plots/hgm. See the README file in that directory for further information. The comparison plots show in three panels the maps of the monthly mean differences, the monthly mean for the current H-Version production and the monthly mean for the reference production, in this case the corresponding D-Version. Red colors denote positive differences (in this case HGM minus D2), blue colors denote negative differences, green indicates nearly zero difference and grey indicates that a difference is not calculated because one or both datasets have no values for that particular location and quantity. Note that the range of difference values on the color bar is +/- 2 standard deviations from the mean. Increasing difference magnitude is shown by colors from nearly white to denser red or blue, the largest differences have the densest colors regardless of magnitude so the Summary Table extreme values should always be checked. The Summary Table above each map provides average, minimum and maximum values for the total global, global land and global water areas, as well as for tropical ($\pm 20^\circ$), north (60° to 90°) polar and south (-60° to -90°) polar regions.

There should be no large artificial features in the monthly mean maps; some that exist because of differences between satellites should nearly disappear in the difference map. The differences between the H-Version and D-Version of the ISCCP products are not expected to be very large. Some differences will be related to revised radiance calibrations for particular satellites and some differences in a few variables will be related to changes in the analysis procedure (noted below). Sudden changes related to a problem with a given satellite will appear in the difference map as an artificial feature. Particularly important pages are 1, 2, 5, 6, 9, 22 and 23. Pages 1 and 2 show total cloud amount and the IR-marginal cloud amount, the latter of which is very stable in space and time. Pages 5 and 6 provide a monitor of the overall IR radiance dependence of the results (see additionally Page 14), while Page 9 monitors the overall VIS radiance dependence (see additionally Page 15). Pages 5-15 together provide a way to monitor changes that might be related to calibration changes or problems by checking the mean cloud and surface properties as well as the extreme cloud types (see discussion of pages 1-3 for D1stats plots in Section 3.3.4). Pages 16-21 monitor the Ancillary inputs to the analysis. Pages 22-23 monitor the number of files from the HGH (monthly mean time of day) comprising the monthly mean results: there should generally be 8 files for Page 22 and a variable number of files for Page 23 but with a regular and distinctive (seasonally varying) geographic pattern.

Page 1: Mean Cloud Amount (%)

Page 2: Mean IR-Marginal Cloud Amount (%)

Page 3: Frequency of Mean Cloud Amount 0-10% (%)

Page 4: Frequency of Mean Cloud Amount 90-100% (%)

Page 5: IR Mean Low Cloud Amount (%)

Page 6: IR Mean High Cloud Amount (%)

Page 7: Mean Cloud (Top) Pressure (mb)

Page 8: Mean Cloud (Top) Temperature (Kelvin)

Page 9: Mean Cloud TAU (Optical Thickness)

Page 10: Mean CA (Cloud Amount) for Cumulus Liquid (%)

Page 11: Mean CA for Status Liquid (%)

Page 12: Mean CA for Cirrus (%)
Page 13: Mean CA for Deep Convective (%)
Page 14: TS (Surface Skin Temperature from Clear Sky (Composite) (Kelvin)
Page 15: RS (Surface Reflectance) from Clear Sky (0-1)
Page 16: Near-Surface Air Temperature (Kelvin)
Page 17: Temperature at 500 mb (Kelvin)
Page 18: Tropopause Temperature (Kelvin)
Page 19: Total Precipitable Water (cm)
Page 20: Ozone Abundance (Dobsons)
Page 21: Snow/Ice Amount (%)
Page 22: Number of Observations (per day – monthly averages available)
Page 23: Number of Daytime Observations (per day – monthly averages available)

One notable change from D-Version to H-Version involves the change of atmospheric temperature-humidity information from TOVS to NNHIRS. The largest changes occur in the polar regions where the tropopause is much lower in altitude and warmer in NNHIRS.

3.4.2 HGG-D1 Comparison Histogram Plots (AS NEEDED for INVESTIGATION)

The HGG-D1 comparison plots are histograms of differences. This IDL plot package is located in comparison_plots/hgg. See the README file in that directory for further information.

3.4.3 HXS-DX Comparison Histogram Plots (AS NEEDED for INVESTIGATION)

The HXS-DX comparison plots are histograms of differences. This IDL plot package is located in comparison_plots/hxs. See the README file in that directory for further information.

3.5 Post-production – EXTENDED PERIOD

These plots are the same as the ones described above in Section 3.4 except that the comparison of the current month of H-production is made to the same month of H-production from the previous year. Differences may be somewhat larger as a result.

3.5.1 HGM-HGM Comparison Map Plots & Summary Table (ALWAYS EXAMINE)

This IDL plot package is located in comparison_plots/hgm. See the README file in that directory for further information.

3.5.2 HGG-HGG Comparison Histogram Plots (AS NEEDED for INVESTIGATION)

This IDL plot package is located in comparison_plots/hgg. See the README file in that directory for further information.

3.5.3 HXS-HXS Comparison Histogram Plots (AS NEEDED for INVESTIGATION)

This IDL plot package is located in comparison_plots/hxs. See the README file in that directory for further information.

3.6 Isccpview

For routine monitoring of production results it is recommended to spot-check HGH/HGM files using isccpview to look for obvious problems such as missing satellites or bad calibration. These

types of problems can usually be spotted by looking at a few of the main variables such as scene, cldamt, n_obs, ts, and rs. Isccpview can also be used to investigate specific issues in any of the data products.

3.6.1 Building isccpview

Go to the isccpview directory and start IDL
Run the *make* batch program:

```
IDL> @make
```

This creates an IDL save file called *isccpview_main.sav* which contains the compiled application. You only have to do this once.

3.6.2 Running isccpview

At the IDL prompt enter:

```
IDL> isccpview
```

or, to start in a particular directory (to save time browsing)

```
IDL> isccpview, dir='/path/to/directory'
```

This restores the pre-built IDL save file *isccpview_main.sav* and runs the program

3.6.3 Opening data files

Use the file browse dialog to select any ISCCP product file. It reads all H-products and most D-products as well. It can also read intermediate products (bx, clrsky etc.) and ancillary products (topo, surfacetype, aerosol etc.)

3.6.4 Viewing maps

Select a variable by clicking one of the buttons, or enter the variable name in the text box if available to bring up a map image of the variable.

3.6.5 Viewing cell contents

When viewing gridded maps, clicking on a map location will bring up a text box with the values of all variables for the selected cell. For some products a cell can be selected by entering the cell number on the variable selection menu.

3.7 Troubleshooting

3.7.1 Error handling system

The main error handler is the function *isccp_error()* which is contained in the *isccp_utils.f90* module. Most steps use this function when they encounter an error to stop the program with an appropriate error message, also causing *prdrun* to stop. In the case of a bug-type error, the

program may exit unexpectedly and in that case the return code from the run script is non-zero and *prdrun* will stop.

3.7.2 Warnings

Some conditions are just things you want to keep aware of, they do not necessarily stop production. These are termed WARNING and are echoed to the console. Some WARNING messages are expected, such as missing satellite input data files. Unexpected WARNING messages should be investigated.

3.7.3 Errors

In the case of an error condition that requires production to stop, the *isccp_error()* function is used to print an ERROR message to the log. The *prdrun* script will then stop.

3.7.4 Investigating

The problem may be either a system problem or a data problem.

In the case of a system problem, where a code exits unexpectedly, the first step is to re-compile with `DEBUG=yes` to check for things like out-of-bounds index. It is sometimes helpful to go into the directory where the program was running and re-start it by hand. When steps exit with errors, they leave all the symbolic links and Fortran units defined, and the input file which contains the arguments to the step will still be present. You can run the step by hand, for example for *d1prod* you can start it like this:

```
d1prod < d1prod.input
```

and watch the output on the screen, or redirect to a file. This is useful if you are inserting print statements into the code (remember to make install), so you don't have to use the whole *prdrun* mechanism to watch the step re-run for an investigation.

The run scripts can also be started manually, by supplying the appropriate arguments. You must be in the appropriate directory to run them. If it is a satellite step, go into the *satpos* directory then start the script from there, supplying "." as the *wkdir* argument to make it run in that directory.

When re-running programs or scripts by hand, remember to first set the environment in the *wkdir* by sourcing the *prdev* script ("`./prdev`" from within the *wkdir*). If you can't start a script most likely this is the problem.

In the case of a data problem, the logs can be examined for unusual contents, and *isccpview* can be used to look at data files and/or intermediate step output files and/or ancillary data files to investigate as needed.

3.7.5 Re-starting

Always check the schedule to make sure production will re-start at the appropriate point. Sometimes you will just need to pick up where it left off, and other times you will need to re-run

one or more steps. You can edit the tasklist file manually, or use *prdsched* with no arguments to read the current tasklist and edit it with the GUI.

Each time a step is re-run the number suffix on the log file name will increment so that they are not overwritten. You can keep them all, or clean out the log directory if you don't want to keep old logs once the problem has been fixed. If you remove them, the number will start over again at zero.

If you are re-running a step, the data files will be overwritten, but it is a good idea to look around and clean them out anyway if they were bad.

Once things have been fixed, and the schedule has been reviewed and adjusted if necessary, restart *prdrun* in the usual way: \$ nohup prdrun &

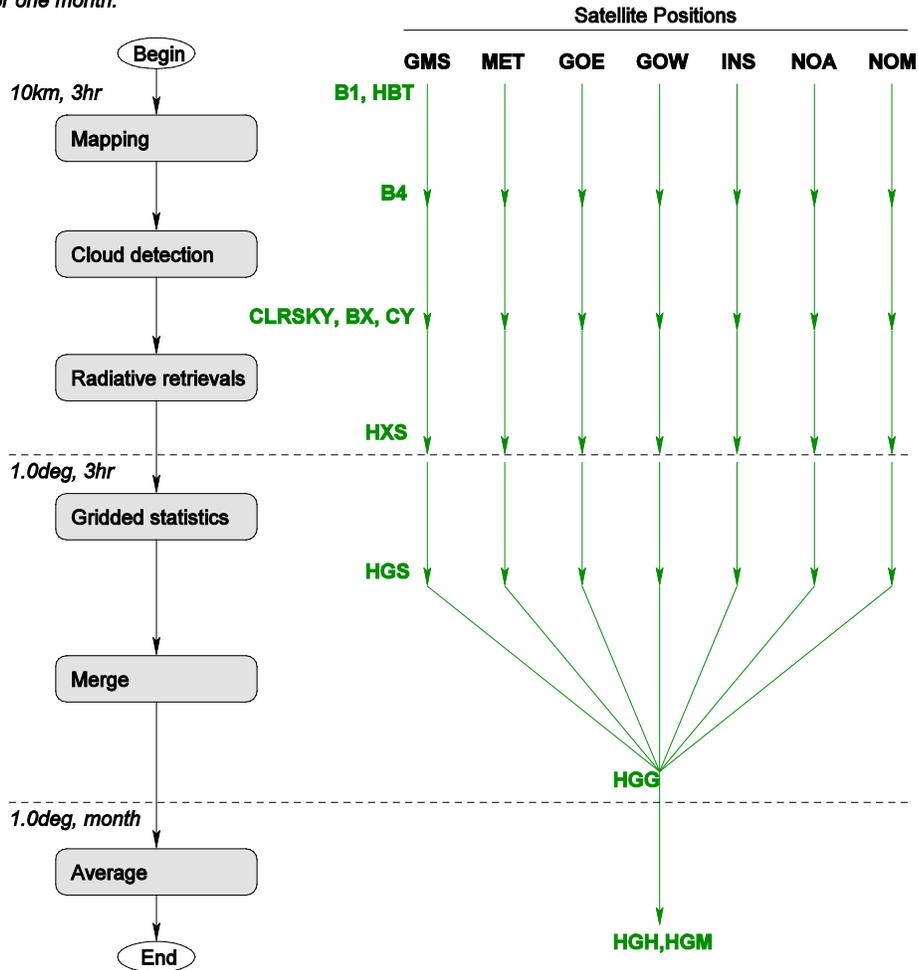
4 System architecture

This section contains a basic description of ISCCP production software and tables. See also *ISCCP_dataflow_diagram.pdf*.

4.1 Production data flow overview

ISCCP Data Flow Overview

For one month:



4.2 Production cycle

4.2.1 Environment variables (prdenv)

The production system relies on shell environment variables for global settings. These variables tell the system where to find codes, tables and data. Environment variables also set various switches that control production. The variables are set via the prdenv script. This script should be copied into each working directory and edited specifically for the desired run. Most of the environment variables are set once at the time of system installation and do not need to be edited for each run. These are described by comments in the prd/bin/prdenv.template file. A few of the environment variables are useful to edit specifically for the desired run. These are:

ISCCP_HXG_COMPRESSION sets the compression level for NetCDF HXG product. This can be 0-9. The recommended value is 4. A value of 0 will turn off compression.

ISCCP_SATPOS_N and ISCCP_SATPOS_LIST together these two control which satpos will be available on the schedule, and in what order they will be processed by prdrun. Changing this does not affect the hierarchy, only the processing order.

ISCCP_STEP_CLEAN controls whether intermediate files are erased after each step completes. This array of flags corresponds to the six steps of production listed in ISCCP_STEP_NAMES. A flag value of 1 will cause prdrun to call the erase.run script after the corresponding step has completed. The erase.run script will remove the intermediate files that were input to the step and therefore no longer needed. A flag value of 0 means nothing will be erased. This is handy for troubleshooting to preserve intermediate outputs for inspection and restarts.

4.2.2 Scheduling GUI (prdsched)

The first time prdsched is called, it uses Sathist to set up the sub-directories for each satpos. Each sub-directory will contain a file {yyyy_mm}.hist which contains a brief summary of the Sathist info for the satpos. The selections made to schedule production are written to a file in the wrkdir call "tasklist". This is a text file which can also be edited by hand.

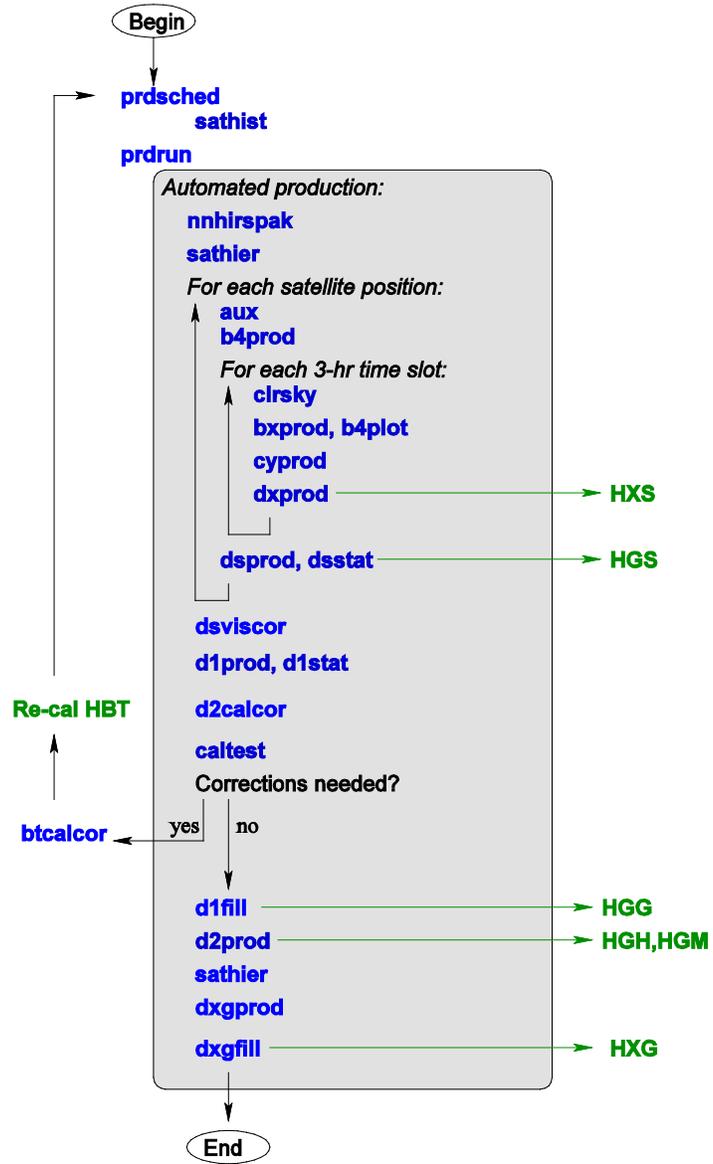
4.2.3 Top-level run script (prdrun)

The prdrun script reads the tasklist and initiates execution of the run script for each task scheduled. If the run script executes successfully, it goes on to run the next step. If the run script exits with a non-zero return code, prdrun will stop. Some steps have pre-processing and/or post-processing tasks associated with them. Prdrun automatically runs these along with the scheduled task. After completion of a step, prdrun scans the step's logfile for the words "WARNING" or "ERROR". Warnings are echoed to the console log. Errors cause prdrun to stop.

The order in which steps are executed is illustrated in the following flowchart of the production cycle. Each step is discussed in further detail in the next section.

ISCCP Production Cycle - H series

For one month:



4.3 Production modes

Production may be run in either calendar mode or sliding window mode. Calendar mode processes one month at a time. Sliding window mode creates a sliding +/- 12 (or other size eg. +/- 17) day window to compute clear-sky maps for each day, and takes considerably more processing time to run. At this time, only calendar mode is to be used for production. In future, if sliding mode is used, the b4link script is provided to setup links to surrounding months to the month being produced so that appropriate b4 data can be found for the sliding window. The production mode is controlled by three environment variables, set in the prdenv script:

ISCCP_RUN_SLIDING: select calendar or sliding mode
ISCCP_RUN_CLAMPBEG: in sliding mode, fixes the begin date
ISCCP_RUN_CLAMPEND: in sliding mode, fixes the end date

4.3.1 Calendar Mode (normal production)

ISCCP_RUN_SLIDING=0

- Clear-sky from calendar month (30d) w/trend cor, like current ISCCP
- 5d: 1-5, 6-10, 11-15, 16-20, 21-25, 26-30
- 15d: 1-15, 16-30
- 30d: 1-30
- Requires 3d in each 5d composite, 4 composites in month
- For months with 31 days, day 31 has no 5-day composite
- For months with 28 or 29 days, there are enough days present to make the last composite (days 26-28 or 26-29)

4.3.2 Sliding Mode (not used)

Open ended example:

ISCCP_RUN_SLIDING=12 (25d window)

ISCCP_RUN_CLAMPBEG=0

ISCCP_RUN_CLAMPEND=0

- Clear-sky from +/-12 days around day being processed, no trend cor
- 5d: d-12 to d-8, d-7 to d-3, d+/-2, d+3 to d+7, d+8 to d+12
- 15d: d+/-7
- 25d: d+12
- Requires 3d in each 5d composite, 4 composites in 25d window

Fixed begin date example:

ISCCP_RUN_SLIDING=12 (25d window)

ISCCP_RUN_CLAMPBEG=1

ISCCP_RUN_CLAMPEND=0

- First 12 days use clear-sky from first 30 days w/trend cor (like calendar)
- Days after that use clear-sky from +/-12 days, no trend cor (like open ended)

Fixed end date example:

ISCCP_RUN_SLIDING=12 (25d window)

ISCCP_RUN_CLAMPBEG=0

ISCCP_RUN_CLAMPEND=1

- Last 12 days use clear-sky from last 30 days w/trend cor (like calendar)
- Days previous to that use clear-sky from +/-12 days, no trend cor (like open ended)

Fixed begin and end dates example:

ISCCP_RUN_SLIDING=12 (25d window)

ISCCP_RUN_CLAMPBEG=1

ISCCP_RUN_CLAMPEND=1

- First 12 days use clear-sky from first 30 days w/trend cor (like calendar)
- Last 12 days use clear-sky from last 30 days w/trend cor (like calendar)
- Days in between use clear-sky from +/-12 days, no trend cor (like open ended)

Typical sliding mode production would be sliding/open.

Fixed begin would be used when processing the beginning dates of a new satellite.

Fixed end would be used when ending a satellite, or when running up to current day.

These sliding window examples have used +/- 12 day window, but other window sizes may be used. The window is always centered on the day being processed, so the central +/- 2 days forms one composite, then additional +/-5 day periods form additional composites beyond that.

4.4 Pre-processing steps

4.4.1 Ancillary data pre-processing (nnhirspak)

The ancillary NNHIRS dataset is pre-processed to calculate brightness temperature profiles, atmospheric emission and extinction. The output is TBNN files that are used later in the dxprod step for retrievals.

Program: nnhirspak.f90
 Input: NNHIRS, AEROSOL, OZONE
 Output: TBNN = <wrkdir>/tbnn/tbnnhirs.yymmddhh

4.4.2 Satellite hierarchy (sathier)

In order to merge the satellites, a satellite hierarchy is created by the sathier step. This hierarchy defines the first, second, third and fourth best choices of satellite for each grid cell, based on the actual satellite locations. This will be used by d1prod to merge the individual 1.0 degree gridded data cells. It is run a second time to create a higher resolution hierarchy for the 0.10 degree pixel-level merge for hxgprod. See Section 5.4 for the description of satellite positions.

Program: sathier.f90
 Input: SATHIST
 Output: SATHIER = <wrkdir>/sathier/yyyy_mm.sathier_1.00 1.0° hierarchy for HGG
 <wrkdir>/sathier/yyyy_mm.sathier_0.10 0.1° hierarchy for HXG
 <wrkdir>/sathier/yyyy_mm.sathier_1.00.png (if IDL is enabled)

4.4.3 Auxilliary data preparation (aux)

The aux step pre-processes the ancillary data for each satellite position and creates maps of the variables to be used in clear-sky step.

Program: auxgeo.f90, geomap.f90 (geostationary), auxpol.f90, polmap.f90 (polar orbiters)
 Input: TOPO, SRFTYP, REFCEO
 Output: AUX =<wrkdir>/aux/sss_t.coasts
 <wrkdir>/aux/sss_t.latlon
 <wrkdir>/aux/sss_t.lndflg
 <wrkdir>/aux/sss_t.srftyp
 <wrkdir>/aux/sss_t.topogr

where sss is the satpos, and t is the type (0=geosat, S,N,A,D for orbiters)

4.5 Post-processing steps

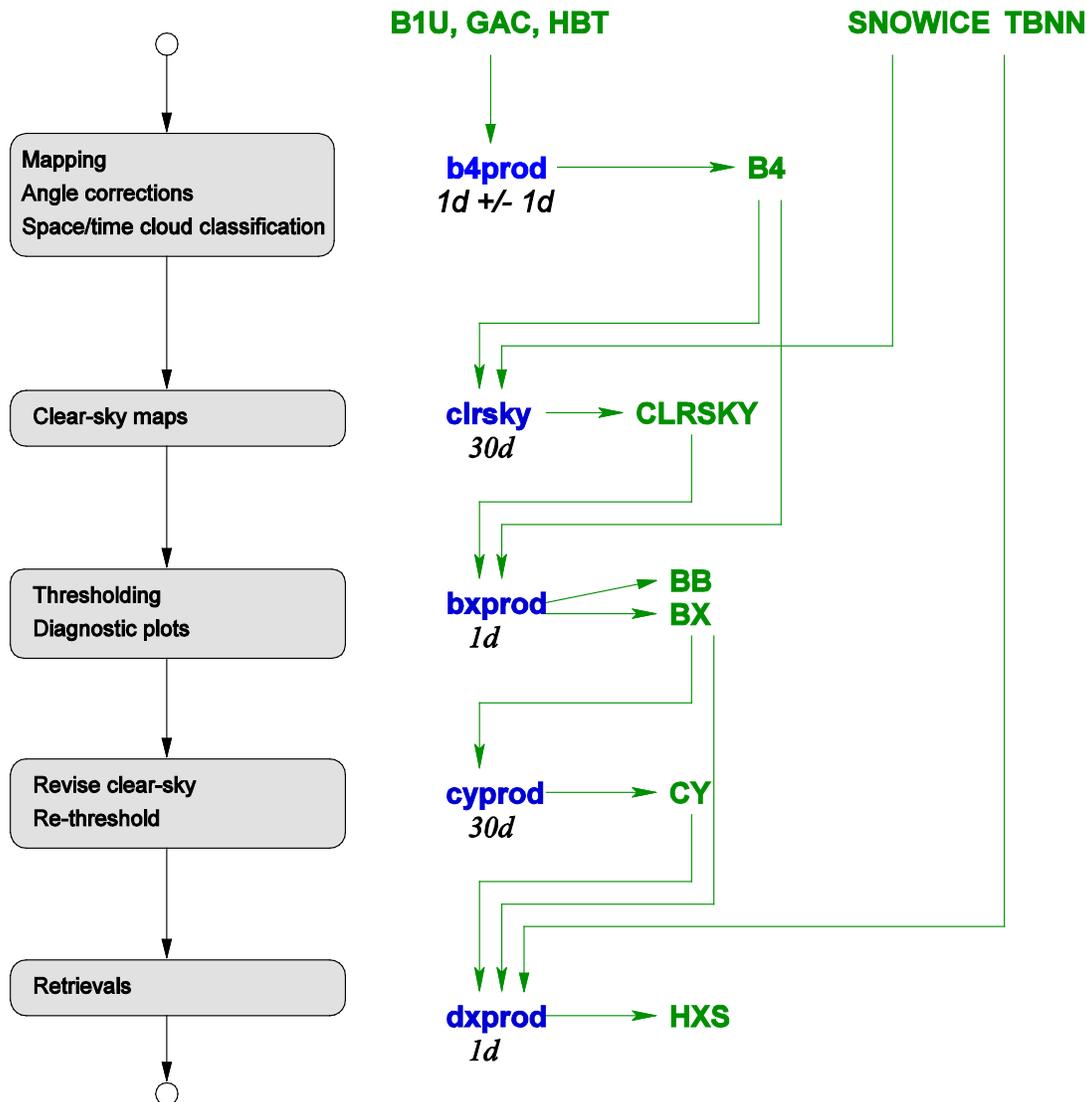
After each step completes, there are several tasks that are automatically performed by the prdrun script. The step log is scanned for errors and/or warnings. Warnings are echoed to the main console, while an error will cause production to stop. If the step was successfully completed, the input files are kept or removed according to the setting of the ISCCP_STEP_CLEAN environment variable. Some steps have post-processing plots that are generated.

4.6 Cloud detection and retrieval steps

There are five main steps that perform the cloud detection and property retrieval for pixel-level products: b4prod, clrsky, bxprod, cyprod, and dxprod. The chained data flow for these steps is illustrated in the following flowchart.

ISCCP Cloud Detection - H series

For one 3-hr time slot, one satellite, one month:

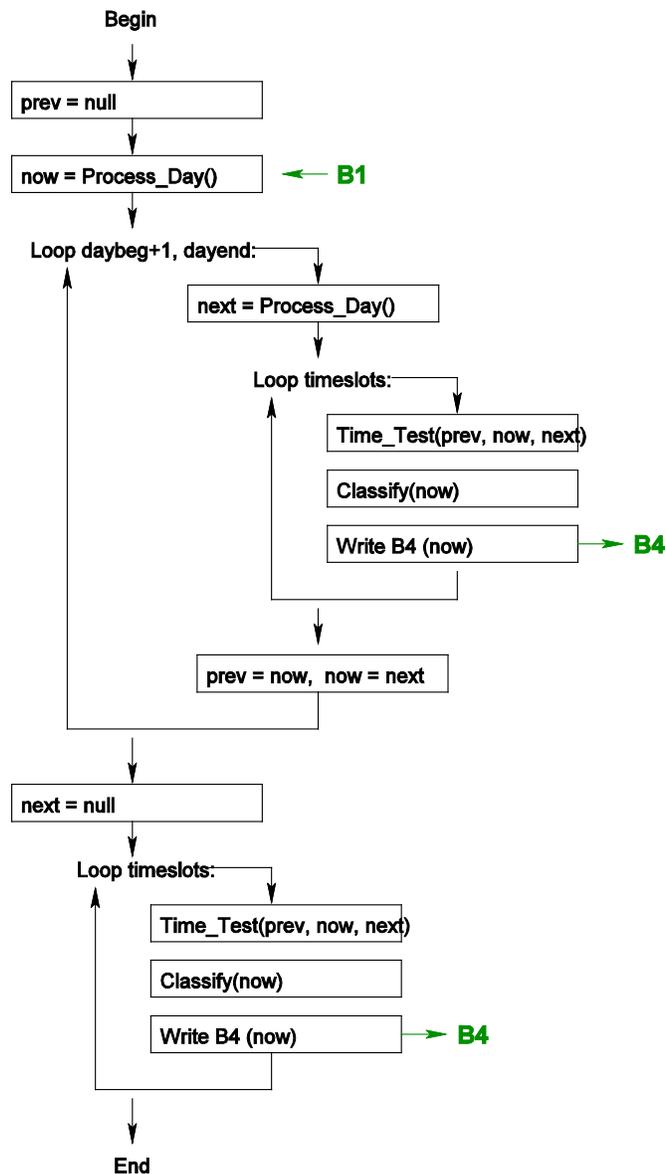


4.6.1 Mapping (b4prod)

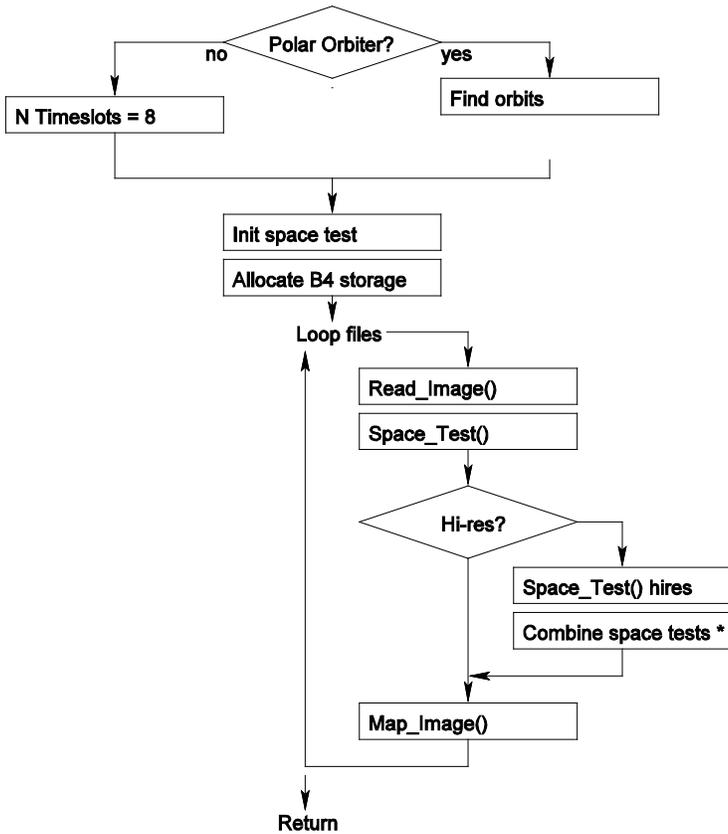
Historically, b4 was the name given to the mapped b3 data. Here it represents the first step of the process, which reads the b1u/gac data, calibrates the radiances, does the space test for cloud detection, maps the data to an i,j location, and does the 3-day time test for cloud detection. The metssp_module is new for H-series and it adjusts the Meteosat sub-satellite point based on external dataset.

Program: b4prod.f90, spacetest_module.f90, metssp_module.f90
Input: B1U, GAC, HBT (NOTE: Stage +/- 1 day around period being processed)
Output: B4

B4PROD



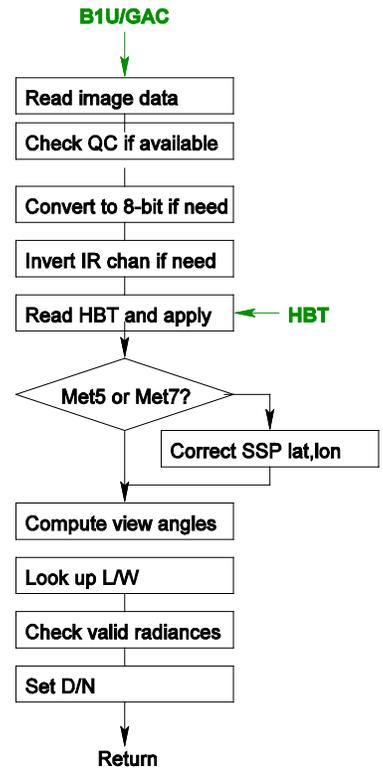
Process_Day(i)



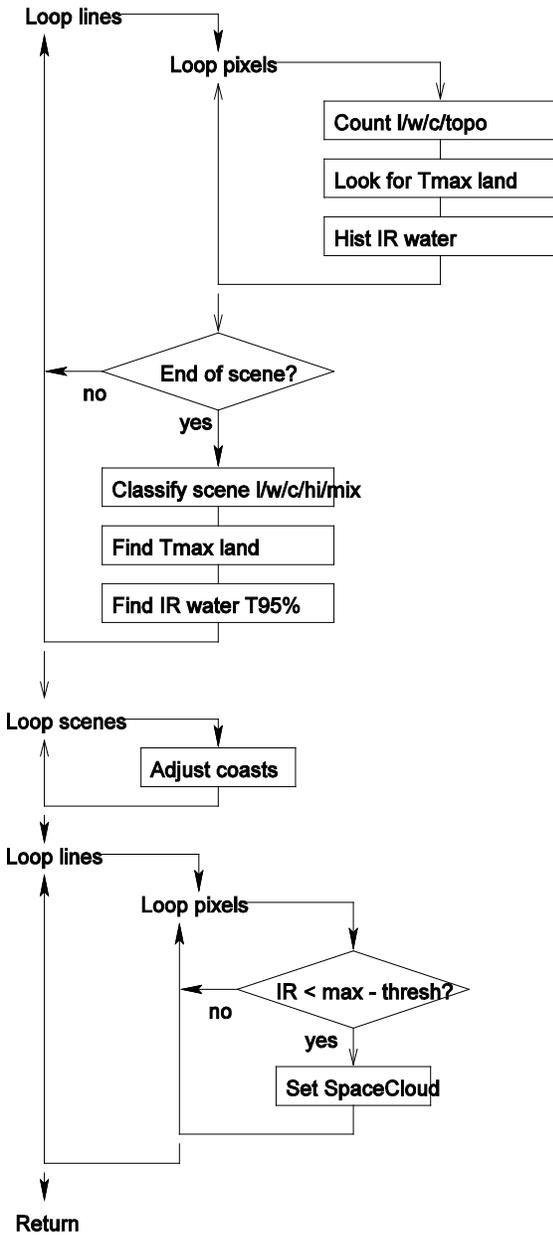
*** Combine space tests**

If low-res sub-scene is mixed surface
 Use high-res result
 Else
 Use low-res result

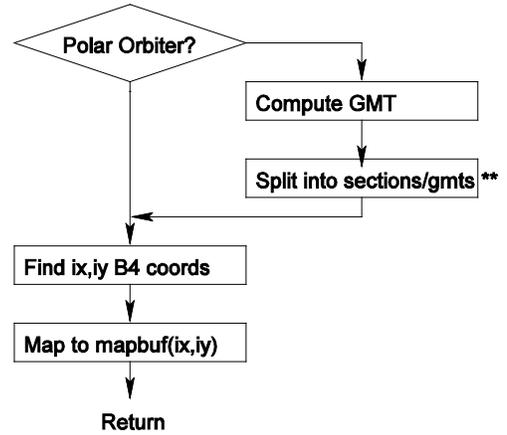
Read_Data()



Space_Test(img)

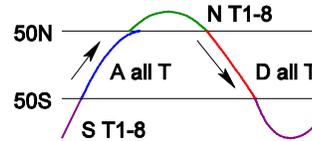


Map_Image(img)

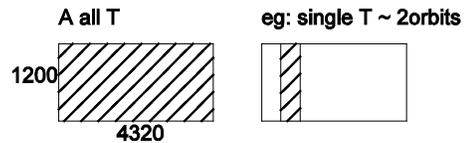


**** Split orbit into sections/gmts**

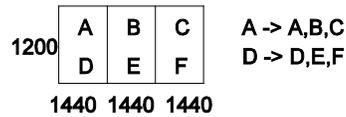
1. Split orbit into S,N,A,D (b4prod)
 N,S -> T1-T8 files
 A,D -> all T in one file



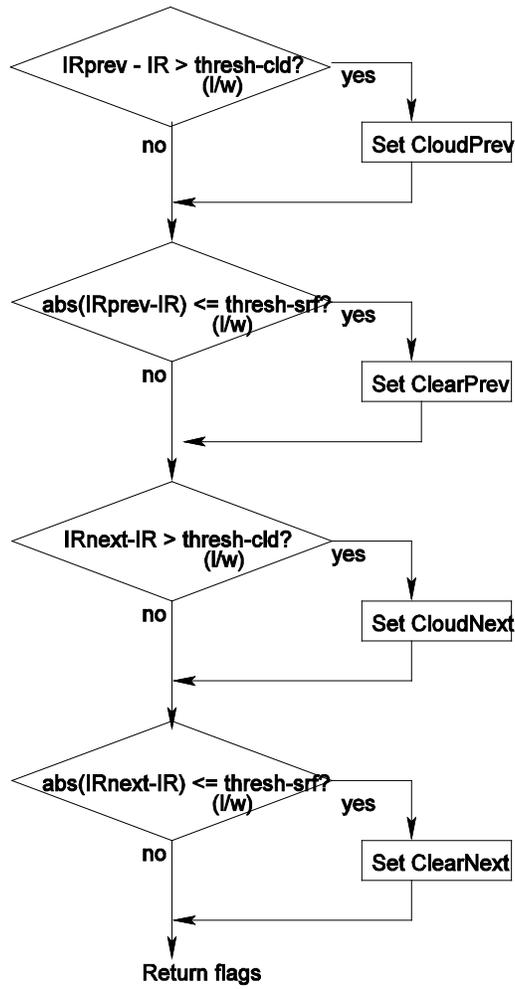
2. Split A,D into T1-T8 files (bxprod) according to scan_nomgmt



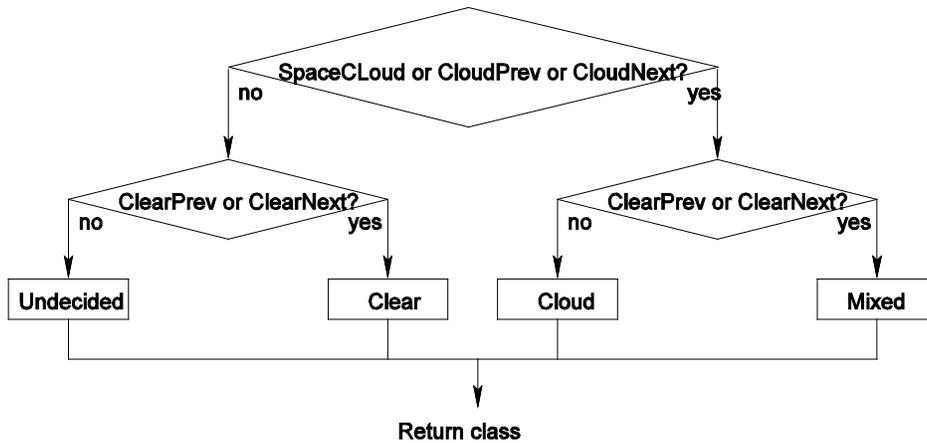
3. Assign A,D to sectors (dxprod) and write them to separate files



Time_Test(prev, now, next)



Classify(img)

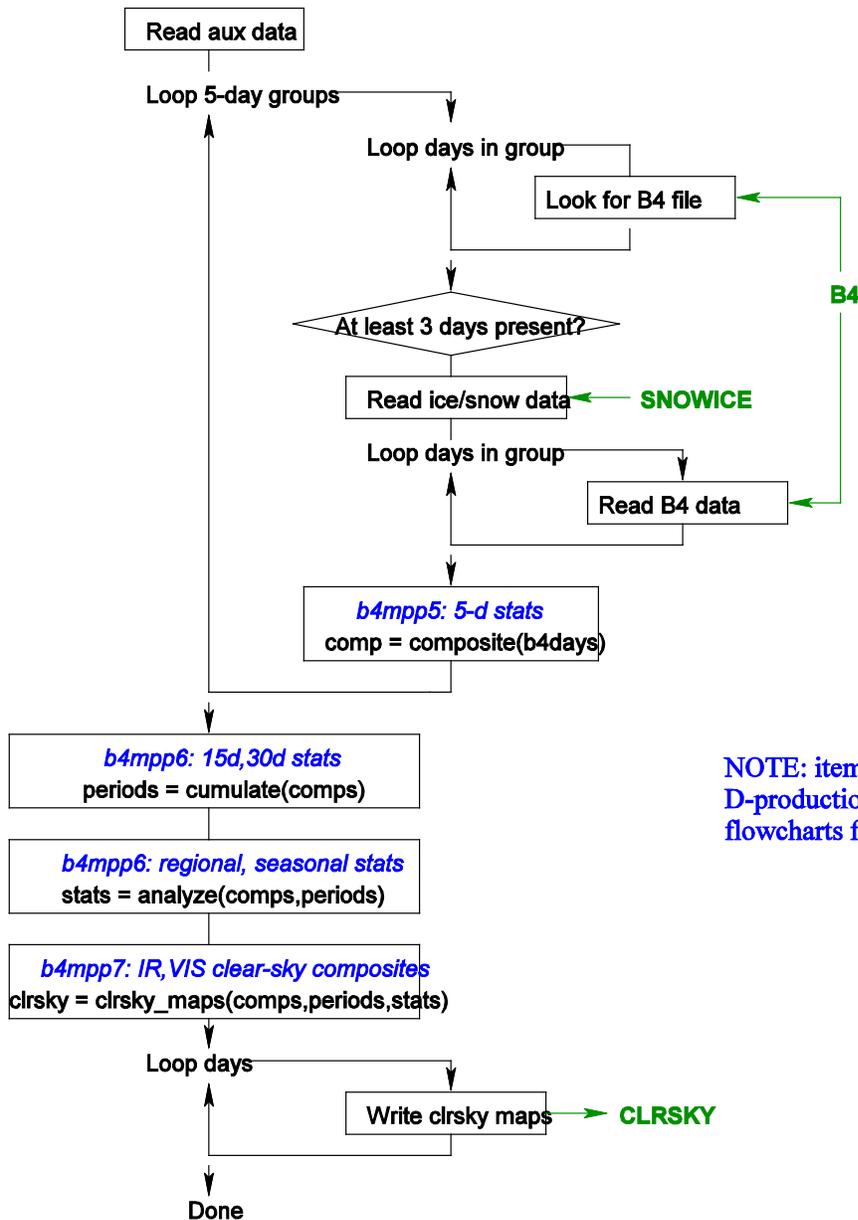


4.6.2 Clear-sky composites (clrsky)

The clrsky step creates the 5-day short term clear-sky composites, and the 15/30 day long term clear-sky composites.

Program: clrsky.f90
 Input: B4, AUX, SNOWICE
 Output: CLRSKY

CLRSKY (Clear-sky composite)



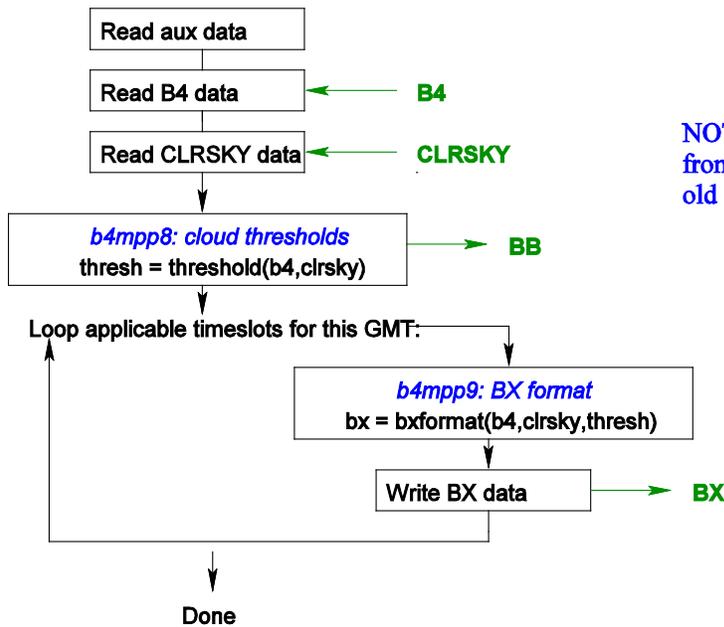
NOTE: items in blue are step names from D-production for cross-referencing to old flowcharts for these sections

4.6.3 Preliminary cloud thresholding (bxprod)

The bxprod step does a preliminary thresholding for cloud detection.

Program: bxprod.f90
 Input: B4, CLRSKY, AUX, SNOWICE
 Output: BX BX image data
 BB Text files containing statistics for b4plot

BXPROD (Threshold, BX format)



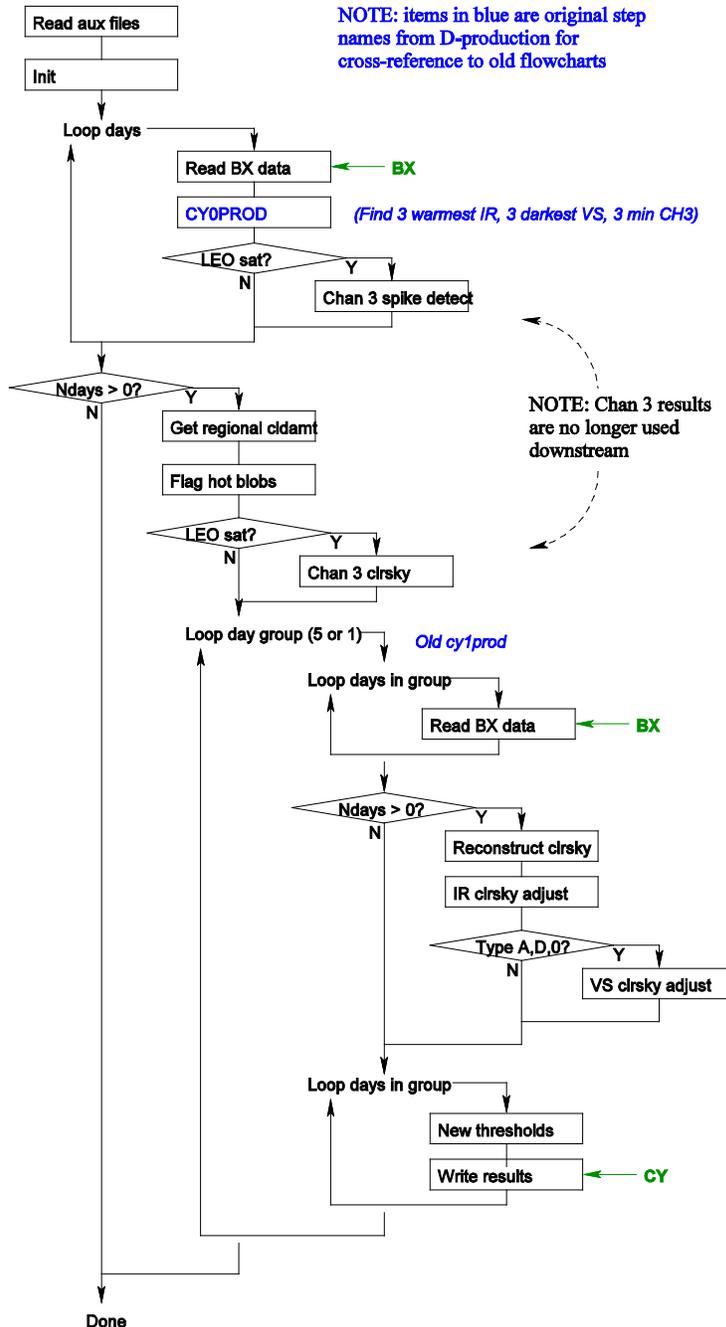
NOTE: items in blue are original step names from D-production for cross-reference to old flowcharts

4.6.4 Revised clear-sky and re-thresholding (cyprod)

The cyprod step analyzes and refines the clear-sky composites and re-thresholds for the final cloud detection result. Note that polar orbiter channel 3 (NIR) results are no longer used, so it does not matter if that channel is available or not.

Program: cyprod.f90
Input: BX, AUX
Output: CY

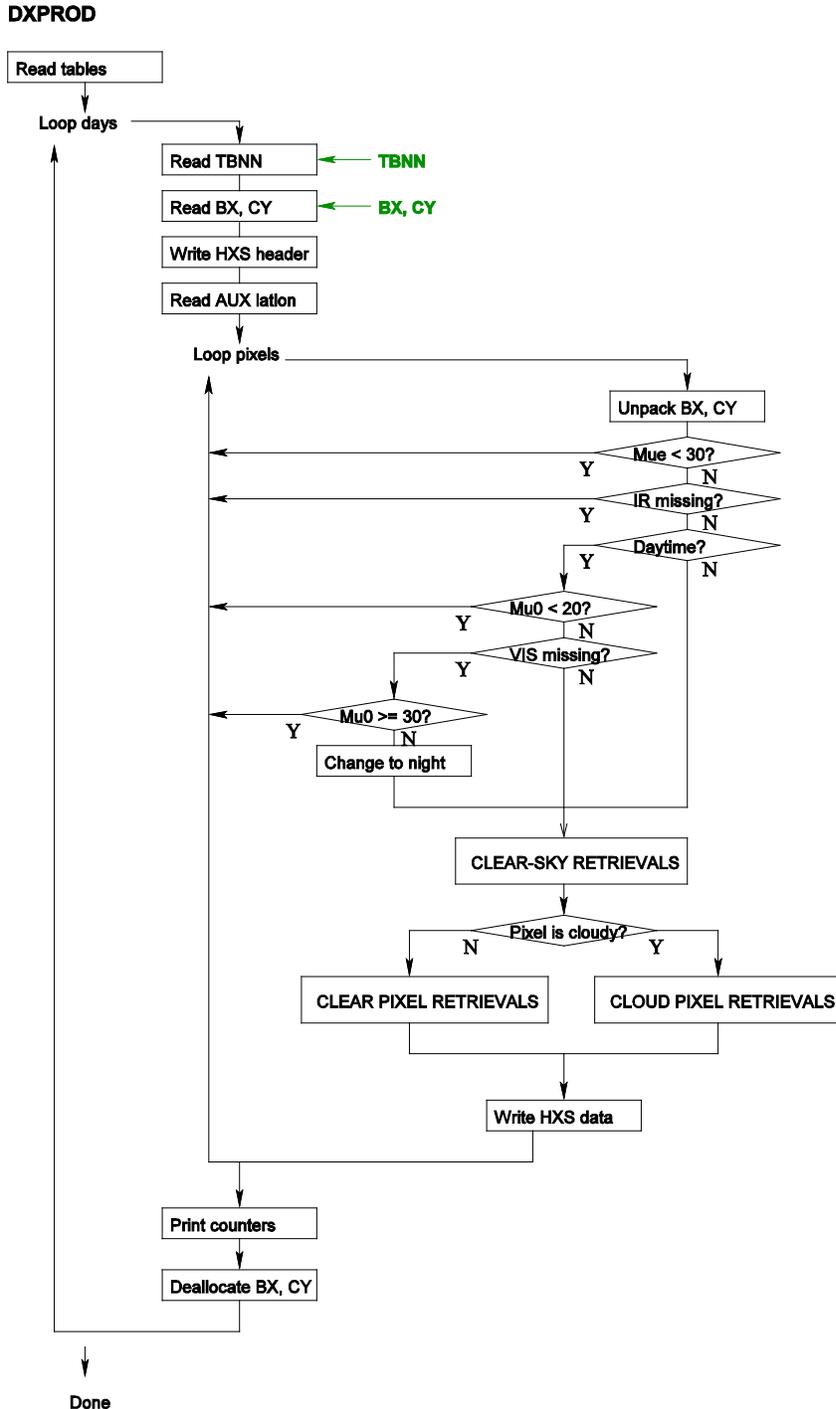
CYPROD



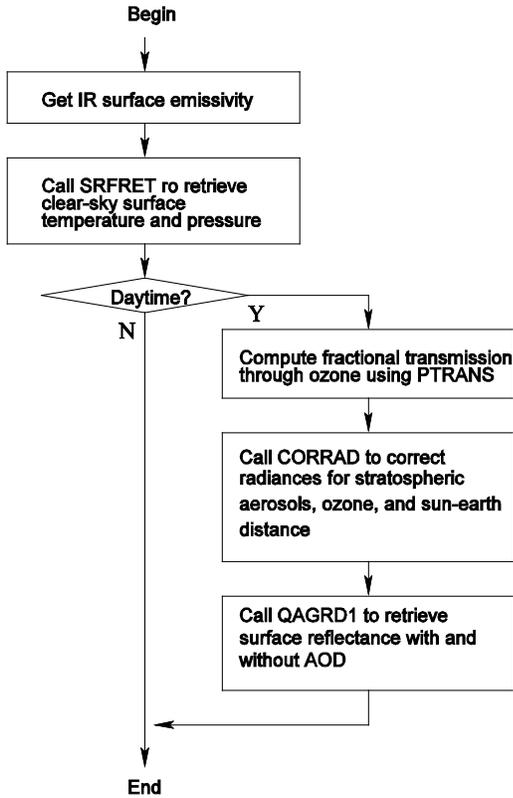
4.6.5 Cloud and surface property retrievals (dxprod)

The dxprod step does the cloud and surface property retrievals, and combines all the results into the HXS product. Coast pixels (defined as being within 20km of opposite land/water type) are removed from this product.

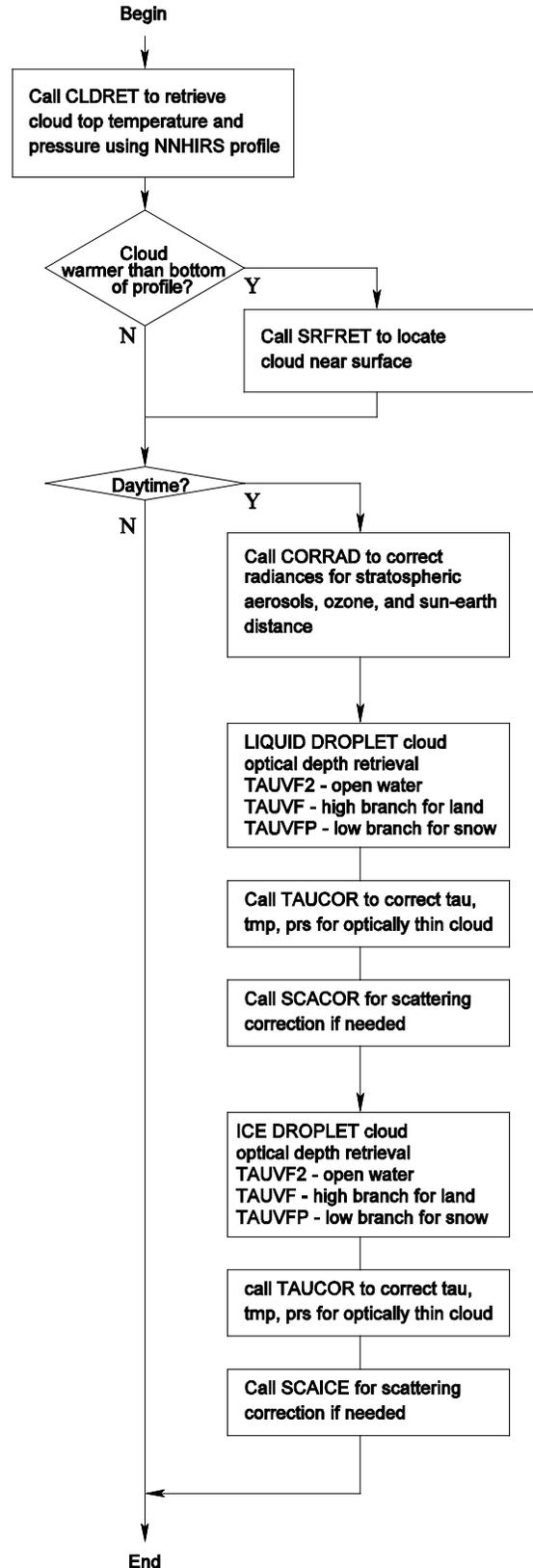
Program: dxprod.f90
 Input: BX, CY, TBNN, AEROSOL, PTRANS
 Output: HXS product



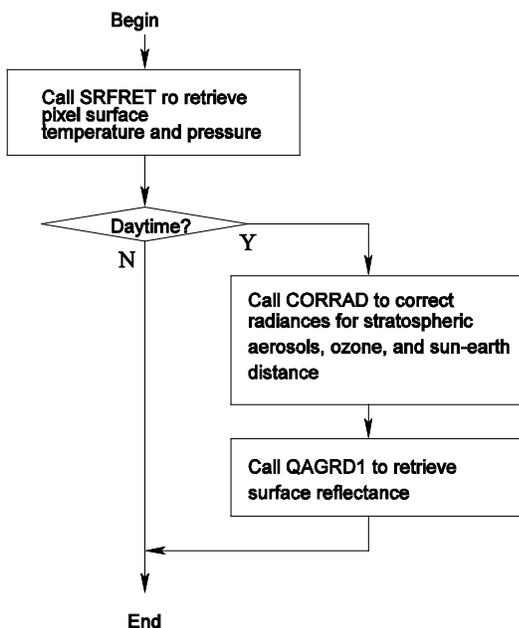
CLEAR-SKY RETRIEVALS



CLOUD PIXEL RETRIEVALS



CLEAR PIXEL RETRIEVALS



4.7 Gridded product steps

4.7.1 Gridding (dsprod)

The dsprod step does the gridding of the pixel data into 1.0 degree equal area map cells, and appends the atmospheric data from NNHIRS. This is essentially the same product as un-filled HGG, only for separate satellites. In ISCCP processing physical quantities are encoded as count values 0-254 using conversion tables. Count value 255 is reserved to indicate “undefined”. The conversion tables for temperature and optical depth are non-linear. Grid cell averaging is done using these encoded counts, except for water path which is averaged in physical values.

Program: dsprod.f90
 Input: HXS, SNOWICE, NNHIRS, OZONE
 Output: HGS
 DSSTAT = <wrkdir>/<satpos>/ds/dsstat.sss.yyyy_mm

4.7.2 VIS-correction (viscor)

For satellites with broad VIS channels the geostationary grid cells are compared to the polar orbiter grid cells where they overlap to compute a VIS correction to be applied during the merge in the next step. Currently this applies only to MET-1 through 7 and MTSAT-1 and 2.

Program: dsviscor.f90
 Input: HGS product
 Output: VISCOR = <wrkdir>/d1/dsviscor.yyyy_mm corrections to be applied in
 d1prod
 <wrkdir>/d1/dsviscor.pairs.yyyy_mm pair stats used to get corrections
 <wrkdir>/d1/dsviscor.scatter.yyyy_mm scatterplot used to get corrections

4.7.3 Merging (d1prod)

The d1prod step merges the satellites by selecting the best one for each grid cell, according to the hierarchy. Snowice, atmosphere and ozone are reported always, even in otherwise empty cells.

Program: d1prod.f90
 Input: HGS (DS), VISCOR, SATHIER, NNHIRS, OZONE
 Output: HGG un-filled
 OVLAP = <wrkdir>/d1/d1ovrlap.yyyy_mm Overlap statistics
 D1STAT = <wrkdir>/d1/d1stat.yyyy_mm Stats for dsd1stat plot

4.7.4 Calibration test

After the merge, the d2calcor step analyzes the overlap stats to compute a calibration correction. The caltest script checks the corrections to see if BTRECAL is necessary. See Section 9.

Program: d2calcor.f90, caltest.pl
 Input: OVLAP
 Output: CALCOR = <wrkdir>/d1/d2calcor.yyyy_mm Calibration corrections
 required

4.7.5 Filling (d1fill)

This is new for H-series. The HGG data is filled using a series of steps. The first step is diurnal interpolation, then daily interpolation, then polar-only spatial replication, then weekly interpolation, and then finally a climatology fill. The climatology fill step is not being used now, but is included here in case it is resurrected in the future.

Program: d1fill_diurnal.f90, d1fill_daily.f90, d1fill_weekly.f90, d1fill_climatology.f90,
d1fill_module.f90
Input: HGG un-filled (NOTE: Stage +/- 2 weeks around period being processed)
Output: HGG product

The HGG (D1) filling process has four parts, each with multiple sub-steps, for a total of 8 procedures:

F1 A = Estimate VIS quantities at night
B = Interpolate VIS quantities overnight
F2 A = Fill missing diurnally
B = Fill missing daily
F3 A = Polar replicate longitude
B = Polar replicate latitude
C = Fill weekly
F4 = Fill climatology

These 8 procedures are implemented in four separate codes: d1fill_diurnal (F1A, F1B, F2A), d1fill_daily (F2B, F3A, F3B), d1fill_weekly (F3C) and d1fill_climatology (F4). Each of these four codes may be run independently, or they may all be run in sequence. The codes share a module file called d1fill_module.f90.

4.7.5.1 d1fill_diurnal.f90

- Requires +/- 7 GMT of original HGG product
- F1A: Estimates VIS at night using VIS/IR – IR method
 - N-cloudy, PC, TC
- F1B: Fills VIS at night by interpolating nearest daytime timeslots
 - Tau, WP
 - N-types (15 pc/tau types)
 - RS-clrsky
- F2A: Fills missing cells by interpolating nearest timeslots if present within +/- 3 hours, both sides having same D/N status
 - Day/night/land/water code
 - N-total
 - N-cloudy & properties
 - N-IR-cloudy & properties
 - N-IRmarg-cloudy & properties
 - N-types (15 pc/tau types)
 - N-IRtypes (3 pc types)

- RS-clrsky

4.7.5.2 d1fill_daily.f90

- Requires +/- 5 days of d1fill_diurnal product
- F2B: Fills missing cells by interpolating nearest full days at same GMT within +/- 5 days, both sides having same D/N status
 - Day/night/land/water code
 - N-total
 - N-cloudy & properties
 - N-IR-cloudy & properties
 - N-IRmarg-cloudy & properties
 - N-types (15 pc/tau types)
 - N-IRtypes (3 pc types)
 - RS-clrsky
 - TS-clrsky
- F3A: Longitude replication – polar (55°+), same variables as F2B, nearest eq cell within max 3 lons having same L/W, same snow/ice (some or none), and eq cell height diff <= 1000m
- F3B: Latitude replication – polar (55°+), same variables as F2B, nearest eq cell within max 3 lats having same L/W, same snow/ice (some or none), and eq cell height diff <= 1000m

4.7.5.3 d1fill_weekly.f90

- Requires +/- 17 days (+/-2 weeks beyond current week) of d1fill_daily product
- F3C: Fills missing cells by getting weekly means (4 days required) and interpolating nearest weeks at same GMT within +/- 2 weeks, both sides having same D/N status
 - Day/night/land/water code
 - N-total
 - N-cloudy & properties
 - N-IR-cloudy & properties
 - N-IRmarg-cloudy & properties
 - N-types (15 pc/tau types)
 - N-IRtypes (3 pc types)
 - RS-clrsky
 - TS-clrsky

4.7.5.4 d1fill_climatology.f90 (not used, included here in case used in future)

- Requires +/- 1 year of d1fill_weekly product
- F4: Fills missing cells by computing 3-yr month-mean at same GMT (month-means are computed, then the three months are averaged but without intermediate conversion back to counts)
 - Day/night/land/water code (assigned according to most common value)
 - N-total
 - N-cloudy & properties
 - N-IR-cloudy & properties

- N-IRmarg-cloudy & properties
- N-types (15 pc/tau types)
- N-IRtypes (3 pc types)
- RS-clrsky
- TS-clrsky

4.7.5.5 Cell Origin

The origin of data in a cell is reported as a code value in the cell_origin variable:

- 0 Original
- 1 Single satellite used in estimating/filling
- 2 Multiple satellites used in estimating/filling

Note that this applies only to the H-data variables, not to the ancillary variables (nnhirs, snowice, ozone) as those are always present and never need to be estimated or filled by this procedure.

4.7.5.6 Fill Type

There are eight bit-flags reported in the fill_type variable indicating the types of filling used in a cell. Multiple flags may be set.

- Bit 1: Estimate VIS at night (number gmts missing reported in fill_gmts variable)
- Bit 2: Interpolate VIS overnight (number gmts missing reported in fill_gmts variable)
- Bit 3: Diurnal interpolation (number gmts missing reported in fill_gmts variable)
- Bit 4: Temporal interpolation (number days missing reported in fill_days variable)
- Bit 5: Longitude replication
- Bit 6: Latitude replication
- Bit 7: Weekly mean interpolation (number weeks missing reported in fill_weeks variable)
- Bit 8: Climatology fill (not used)

4.7.5.7 Notes on filling procedures

1. Estimated-VIS is only applied if nighttime CF > 0 to start with (as was done in D2), otherwise would have a CF reported, but no properties (cannot adjust the properties as there is none reported at all because CF was 0). Minimum value of PC, TC is limited at PTROP, TTROP.

2. The cloudy pixel counters N-cloudy, N-IR-cloudy and N-IRmarg-cloudy are converted to cloud fraction (divide by N-total) before interpolating, and then converted back to an integer number of pixels by multiplying the interpolated cloud fraction by N-total.

3. The cloud type counts N-types and N-IRtypes are converted to proportional cloud fraction (divide by N-cloudy or N-IR-cloudy) before interpolating, and then converted back to an integer number of pixels by multiplying the interpolated proportional fraction by N-cloudy or N-IR-cloudy. If the sum of the cloud types does not equal N-cloudy or N-IR-cloudy (due to round-off) then the types are adjusted by adding/subtracting from the type with the largest number of pixels.

4. All water-path variables are converted to physical (tau) value before interpolating, and then converted back to counts.

5. When interpolating cloud properties, if either previous-full or next-full is undefined (due to CF=0) then replicate the cloud properties from the side that is defined.
6. Cloud properties are only filled if N_cloudy > 0. Similarly, IR cloud properties are only filled if N_IR_cloudy > 0, and IR marginal cloud properties are only filled if N_IRmarg_cloudy > 0.
7. Interpolating between day and night cells results in the filled cell being labeled night, and the day-only properties for IR cloudy and IR marginal cloudy are not filled (Tau_IR, WP_IR, Tau_IRmarg, WP_IRmarg).
8. Diurnal interpolation may use VIS-filled or original cells. Daily interpolation may use diurnally filled, VIS-filled, or original cells. Weekly interpolation and climatology fill use only VIS-filled or original cells. Lat/lon replication may use any cells except those already replicated.
9. The daily, weekly, and climatology fill is done only on empty cells. This means that if the VIS-filling (VIS interpolation over night) failed for the night variables in a full cell, they will remain unfilled for that cell.
10. Snowice, atmosphere and ozone are always present, even in otherwise empty cells, and so are not subject to any filling procedures.

4.7.6 Averaging (d2prod)

The d2prod step does the monthly averaging, first by timeslot over the month (d2proda.f90), and then over all eight timeslots (d2prodc.f90). There used to be a d2prodb.f90 that applied some adjustments to night-time vis data, but that is now done during the d1fill, so there is no longer a d2prodb. The two parts are run via a single d2prod run script. In ISCCP processing physical quantities are encoded as count values 0-254 using conversion tables. The conversion tables for temperature and optical depth are non-linear. Monthly averaging is done using these encoded counts, except for water path which is averaged in physical values. Snowice, atmosphere and ozone are always present, other variables are averaged only over full cells.

Program: d2proda.f90, d2prodc.f90
 Input: HGG filled, SATHIER
 Output: HGH product
 HGM product
 <wrkdir>/d2/d2satpos.yyyy_mm Sat positions found in data, used for
 metadata

4.8 Global pixel-level product (dxg, dxgfill)

This is new for H-series. It is a globally merged pixel-level dataset that is filled with replication. The HXG product contains only those HXS quantities that are common to all satellites.

Program: dxgprod.f90, dxgfill.f90, dxgfill_module.f90
 Input: HXS, SATHIER (NOTE: Stage +/- 2 timeslots around period being processed)
 Output: HXG product

4.9 Code organization

The system software is in three directories: bin (scripts), lib (misc), src (fortran). The locations of everything are specified via environment variables in the bin/prdenv script. Each Fortran program is run via a perl script which sets parameters and defines input/output files. The automated production system is controlled by a Perl/Tk scheduling GUI (prdsched) and a Perl controller script (prdrun). These scripts interface with the Sathist modules to obtain parameters for individual satellites.

4.9.1 prd/bin

Contains Perl wrapper scripts for each Fortran program. The scripts check environment variables, set parameters, define input and output file access, execute the Fortran program and check return codes. Also contains scripts for scheduling (prdsched) and running (prdrun) production and IDL scripts.

Wrapper scripts typically (but not always) have the same name as the Fortran program they run, with the suffix “.run”. These are (alphabetical order):

aux.run	Runs auxgeo (geostationary) or auxpol (polar orbiter)
b1uqc.run	Runs b1uqc (manual step)
b4plot.run	Generates IDL plots of B4 radiances and clear-sky
b4prod.run	Runs b4prod
btcalcor.run	Runs btcalcor to create new HBT (manual step)
bxprod.run	Runs bxprod
clrsky.run	Runs clrsky
crimgl.run	Runs crimgl (for B3 data)
cyprod.run	Runs cyprod
d1fill_climatology.run	Runs d1fill_climatology
d1fill_daily.run	Runs d1fill_daily
d1fill_diurnal.run	Runs d2fill_diurnal
d1fill_weekly.run	Runs d1fill_weekly
d1prod.run	Runs d1prod
d2calcor.run	Runs d2calcor
d2proda.run	Runs d2proda
d2prodc.run	Runs d2prodc
d2prod.run	Combined run of d2proda and d2prodc
dsd1stat.run	Runs dsd1stat
dsprod.run	Runs dsprod
dsviscor.run	Runs dsviscor
dxgfill.run	Runs dxgfill
dxgprod.run	Runs dxgprod
dxprod.run	Runs dxprod
initial_hbt.run	Runs b1uhbt (geostationary) or gachbt (polar orbiter) (manual step)
nnhirspak.run	Runs nnhirspak
sathier.run	Runs sathier

In addition, there are some other Perl scripts that are used by the automated system:

caltest.pl	Analyzes results of d2calcor to determine calibration corrections
erase.run	Post-process removal of empty files created by a step
sathist.pl	Interface to Sathist modules

Scripts that are initiated manually to control production are:

parallel_prdrun	Initiate production using forked processes
prdrun	Initiate production normally
prdsched	Scheduling GUI
prdenv.template	Template for prdenv script to set environment variables
getdate.pl	Command line utility to convert calendar date to project date
b4link.pl	Sets up links to surrounding month b4 data, for sliding mode
d1link.pl	Sets up links to surrounding month d1 data, for d1 fill steps
calcoefs_update.pl	Example of a script that may be used to automatically update the ABS coefficient files after a RECAL procedure (see Section 9).

Also in the prd/bin directory are the IDL scripts that generate plots. These are in the idl sub-directory:

idl/	
b4plot_geo.pro	b4plots for geostationary satellites
b4plot_pol.pro	b4plots for polar orbiters
dstat_plot.pro	Stats plots for HGS and HGG
sathier_plot.pro	Satellite hierarchy plot
b1uqc_plot.pro	b1uqc plots (run manually as needed)
closemulti.pro	used by b1uqc_plot
getcolor.pro	used by b1uqc_plot
plotsym.pro	used by b1uqc_plot
savemulti.pro	used by b1uqc_plot
setlineplot.pro	used by b1uqc_plot

This prd/bin directory also contains the Fortran executables when the system is built. Once the Fortran in prd/src is compiled, a “make install” will copy the executables to the prd/bin directory and “make realclean” removes them (“make clean” only cleans the prd/src directory).

4.9.2 prd/lib

Contains utility modules used by the scripts:

Dates.pm	Date formatting and conversion
Filenames.pm	Filename generation for input/output data
Numbers.pm	Number formatting
Utils.pm	Misc. utilities to check script args and handle log files
taskfile.pl	Read/write taskfile for prdsched

Also contains the Sathist modules:

Sathist.pm	Sathist wrapper, called by run scripts
Satevents.txt	Satellite events/locations tagged by date
Sat.pm	Satellite descriptions
Satpos.pm	Satellite position info
Spc.pm	Satellite processing centers
sat_print.pl	Handy utility to print the contents of Sat.pm in a more readable format

Also here in prd/lib are the NetCDF format descriptor (CDL) files which are used to define the metadata and formats for the products. These files are in the formats sub-directory.

formats/

```

hgs.cdl
hgg.cdl
hgh.cdl
hgm.cdl
hxg.cdl
ancil/
    aerosol.cdl
    nnhirs.cdl
    ozone.cdl
    snowice_0.25.cdl
    snowice_1.00.cdl
    surfacetype.cdl
    topo.cdl

```

There is no CDL descriptor file for HXS because that is the only product that is not in NetCDF format. Refer to dxdata_module.f90 for native format description of HXS.

An alternate version of the CDL descriptor files for H products is in the sub-directory “h_products_with_comments_attribute”. The files in this sub-directory have the “comment” attribute filled in with explanations of how to re-map the equal-area grid and how to derive some additional quantities that would be useful for a data user to know. These CDL files are not being used because there is a bug in NetCDF which renders them un-readable when viewed with ncdump. This was reported to Unidata and bug ticket NCF-335 was created 22 Jun 2015. For status of the bug ticket see: <https://bugtracking.unidata.ucar.edu/browse/NCF-335>

4.9.3 prd/src

Contains the Fortran programs for each processing step and associated input/output modules. Grouped by functionality, these are:

Equal area grid

eqgrid_module.f90	Equal area grid info read/write
tables/equsqu.f90	Compute equal area grid info

Ancillary data and tables

ancil/	Ancillary data NetCDF conversion
aerosol_module.f90	AEROSOL read/write
nnhirs_module.f90	NNHIRS read/write
tbnn_module.f90	TBNN read/write
nnhirspak.f90	NNHIRS -> TBNN
ozone_module.f90	OZONE read/write
snowice_hires_module.f90	SNOWICE read/write
surfacetype_module.f90	SURFACETYPE read/write
topo_module.f90	TOPO read/write
auxgeo.f90	Prepare mapped ancil - geostationary
auxpol.f90	Prepare mapped ancil - polar orbiter
aux_module.f90	Mapped ancil read/write

Calibration

satable_module.f90	Radiance/temperature conversion table read
btdata_module.f90	HBT calibration table read
b1uhbt.f90	Create initial version of HBT (nominal) for geostationary
gachbt.f90	Create initial version of HBT (nominal) for polar orbiter
btcalfcor.f90	Create new version of HBT calibration tables

New satellite setup

sat_setup/*	Initial setup procedures for new satellites, including PTRANS, RADIUS and REFGEO creation (see Section 5), and NORM, ABS and SATTABLE setup (see Section 7).
-------------	--

Radiance data - B3 or B1U/GAC

satdata_module.f90	Data structure for satellite data
b3data_module.f90	B3 ingest subroutines
b3_Dmodule.f	B3 declarations
b3_kind.f	B3 data type definitions
b3read_be.f	B3 read/unpack subroutine, big-endian
b3read_le.f	B3 read/unpack subroutine, little-endian
crimgl_be.f	B3 QC main program, big-endian
crimgl_le.f	B3 QC main program, little-endian
badimg.f	B3 function to check QC results
b1uqc.f90	B1U/GAC QC main program
b1uqc_module.f90	B1U/GAC QC declarations and functions
b1data_module.f90	B1U ingest subroutines
b1u_kmod/*	Directory contains Ken Knapp's B1U read/unpack/nav code
solzen_module.f90	Solar zenith angle and azimuth for B1U
gacdata_module.f90	GAC ingest subroutines
gac_clavrx/*	Directory contains CLAVRX GAC read/unpack code
metfull_module.f90	Sample Meteosat full disk ingest subroutines (for G.Seze)

Map projection

geomap.f90	Geostationary map projection
polmap.f90	Polar orbiter map projection
metssp_module.f90	SSP corrections for MET5 and MET7

Algorithm steps – Pixel level HXS/HXG products

b4data_module.f90	B4 data read/write
spacetest_module.f90	B4 space test
b4prod.f90	Mapping, space/time tests

clrsky_module.f90	CLRSKY data read/write
clrsky.f90	Clear-sky algorithm

bxdata_module.f90	BX data read/write
bxprod.f90	Threshold

cydata_module.f90	CY data read/write
cyprod.f90	Revise clear-sky, re-threshold

dxdata_module.f90	DX/HXS read/write
dxprod.f90	Retrievals - HXS product
dxgdata_module.f90	HXG read/write
dxgprod.f90	HXG product
dxgfill_module.f90	HXG read/write
dxgfill.f90	HXG filling

Gridding - HGS product

dsdata_module.f90	HGS read/write
dsprod.f90	HGS product

Merging – HGG product

sathier.f90	Satellite hierarchy
dsviscor.f90	VIS correction
d1data_module.f90	HGG read/write
d1prod.f90	HGG product
d1fill_module.f90	HGG filling utils
d1fill_diurnal.f90	HGG diurnal fill
d1fill_daily.f90	HGG daily fill
d1fill_weekly.f90	HGG weekly fill
d1fill_climatology.f90	HGG climatology fill

Monthly means - HGH/HGM products

d2data_module.f90	HGH/HGM read/write
d2proda.f90	HGH product
d2prodc.f90	HGM product

Calibration correction

d2calcor.f90	Analyze overlap stats for calibration correction
Stats plot	
dsd1stat.f90	IDL stats/plots
Utilities	
isccp_nc4.f90	NetCDF4 read/write utils
isccp_utils.f90	Things common to all production steps
isccp_oldutils.f	Used by B3 codes
tables/*	Create static tables that reside in prd/tables, including CNTTAB, TABCNT, ANGCOR and EQUAL
Compilation	
Makefile.template	Sample Makefile

4.10 Data read/write modules

For each type of data there is a corresponding module which contains read/write functions, and other utilities related to that data.

4.10.1 Equal area grid

EQUAL eqgrid_module.f90, tables/equsqu.f90

4.10.2 Auxilliary data

AUX aux_module.f90

4.10.3 Ancillary data

AEROSOL aerosol_module.f90
 NNHIRS nnhirs_module.f90, tbnn_module.f90
 OZONE ozone_module.f90
 SNOWICE snowice_hires_module.f90
 SURFACETYPE surfacetype_module.f90
 TOPO topo_module.f90

4.10.4 Satellite data

B1U b1data_module.f90, b1u_kmod/*
 GAC gacdata_module.f90, gac_clavrx/*
 HBT btdata_module.f90

4.10.5 Intermediate step output

B4 satdata_module.f90, b4data_module.f90
 CLRSKY clrsky_module.f90
 BX bxdata_module.f90
 CY cydata_module.f90

4.10.6 Products

HXS	dxdata_module.f90
HXG	dxgdata_module.f90
HGS	dsdata_module.f90
HGG	d1data_module.f90
HGH/HGM	d2data_module.f90

4.11 Utilities

Common utilities	isccp_utils.f90
NetCDF interface	isccp_nc4.f90
Old still used	isccp_oldutils.f
Old still used	b3_Dmodule.f
Old still used	b3_kind.f
Table creation	tables/*

4.12 Tables (prd/tables)

The contents of prd/tables are data and lookup tables used in production which are created once and do not change, listed alphabetically by table name. These are all provided in the distribution. If the file creation program is available it is in prd/src/tables directory. NOTE that the B1 refgeo files need to be reviewed for accuracy and completeness. The filename of the refgeo file used for a particular satellite is specified in lib/Sat.pm (see Section 5.8).

Table name	Description	Used by	Created by
angcor4.data	Angle corrections	b4, clrsky, bx, cy	angcor4.f90
aveab9.tab42	Absorption coefs	nnhirspak	
aveabn9.tab4	Absorption coefs	nnhirspak	
cnttab4.data	Convert counts to physical values	nnhirspak,b4,clrsky,bx,cy,dx,ds dsviscor,d1,d1fill,d2a,d2c dsviscor,dxgfill	cnttab4.f90
equal_0.10.data	0.1° eq/sq grid	aux,b4	equsqu.f90 (from TOPO and SRFTYP)
equal_0.25.data	0.25° eq/sq grid	aux,bx,clrsky	equsqu.f90 (from TOPO and SRFTYP)
equal_0.50.data	0.5° eq/sq grid		equsqu.f90 (from TOPO and SRFTYP)
equal_1.00.data	1.0° eq/sq grid	aux,dx,ds,d1,d1fill,d2a,d2c sathier,dsviscor,dsd1stat dxg,dxgfill,nnhirspak	equsqu.f90 (from TOPO and SRFTYP)
equal_2.00.data	2.0° eq/sq grid		equsqu.f90 (from TOPO and SRFTYP)
equal_2.50.data	2.5° eq/sq grid	ds,d1,d1fill,d2a,d2c sathier,dsviscor,dsd1stat dxg,dxgfill (2.5 grid option)	equsqu.f90 (from TOPO and SRFTYP)
equal_5.00.data	5.0° eq/sq grid		equsqu.f90 (from TOPO)

			and SRFTYP)
glntab.data	Glint tables	bx	glint.f
hxs_variables.txt	Titles,units,format for HXS variables	dx	Created by hand.
iscaco2.data	Scattering correction	dx	
lndocn.tables	Land and ocean models	bx,cy	
m_ictauHlbrch.dat	Tau retrieval ice cloud, land high branch	dx	
m_ictauHlbrch_aodocn.dat	Tau retrieval ice cloud, ocean	dx	
m_ictauLObrch.dat	Tau retrieval ice cloud, land low branch	dx	
m_wtrtauHlbrch.dat	Tau retrieval liquid cloud, land high branch	dx	
m_wtrtauHlbrch_aodocn.dat	Tau retrieval liquid cloud, ocean	dx	
m_wtrtauLObrch.dat	Tau retrieval liquid cloud, land low branch	dx	
met-ind.met7.+057.0.lat met-ind.met7.+057.0.lon	Lat/lon for Meteosat full disk	aux, b4 (metfull option)	G.Seze
myclr_sky_aodcoarse.dat	Ocean clear-sky model	dx	
myclr_sky_aodfine.dat	Ocean clear-sky model	dx	
myclr_sky_aodfine_ocn.dat	Ocean clear-sky model	cy	
M5_SSP_Collection.dat	Met-5 actual SSP	b4	A.Lattazio Eumetsat
M7_SSP_Collection.dat	Met-7 actual SSP	b4	A.Lattazio Eumetsat
noa9ch3.radtmp	Convert radiance to temperature	cy	

noa9ch4.radtmp	Convert radiance to temperature	dx,tovpak3	
NOTE regarding “ptrans” files below: if a new satellite is added to the system, a corresponding ptrans file must be generated using the response function. See section 5.7.			
ptrans/ptrans_sssnn.data (one file per sat “sssnn”)	Ozone transmission	dx	
ptrans/input/linefile_only_O3_0.bin	Absorption optical depths	ptrans	
ptrans/input/max_min_mmin_spectra.txt	Solar irradiance	ptrans	
ptrans/input/response_sssnn.txt	Response function	ptrans	
raddif_strtaod_nosrf.dat	Stratospheric aerosol correction	dx	
NOTE regarding “refgeo” files below: The filename of the refgeo file used for a particular satellite is specified in lib/Sat.pm. Each satellite that is pre-mapped by the SPC must have a corresponding refgeo file. See section 5.8.			
refgeo.cin	Insat B3 lat/lon	aux	
refgeo.cme	Meteosat B3 lat/lon CME SPC	aux	Copy of refgeo.esa
refgeo.esa	Meteosat B3 lat/lon ESA SPC	aux	refgeo.esa.f
refgeo.eum	Meteosat B3 lat/lon EUM SPC	aux	Copy of refgeo.esa
refgeo.msg	MSG B3 lat/lon	aux	
refgeo.mti	Meteosat over Indian ocean B3 lat/lon	aux	
refgeo495x495.msg	Meteosat B3 lat/lon	aux	
refgeo618x618.msg	Meteosat B3 lat/lon	aux	
refgeoB1.met	Meteosat B1 lat/lon	aux	See Section 5.8
refgeoB1.msg	MSG B1 lat/lon	aux	See Section 5.8
refgeoB1.mts	MTS B1 lat/lon	aux	See Section 5.8
refgeo_new.esa	Meteosat B3 lat/lon	aux	
SATTABLE.ALL	Thermal channel	b1uhbt, gachbt, btcalcor	See Section

	tables		7.2.4
surfacetype_eqarea_0p25.nc	SURFACETYPE product 0.25°	aux	srftyp2nc.f90
surfacetype_eqarea_0p5.nc	SURFACETYPE product 0.5°	aux	srftyp2nc.f90
surfacetype_eqarea_1p0.nc	SURFACETYPE product 1.0°	aux	srftyp2nc.f90
surfacetype_eqarea_2p0.nc	SURFACETYPE product 2.0°	aux	srftyp2nc.f90
surfacetype_eqarea_2p5.nc	SURFACETYPE product 2.5°	aux	srftyp2nc.f90
tabcnt4.data	Convert physical values back to counts	b4,bx,clrsky,cy,dx,ds d1,d1fill,d2a,d2c dxgfill,nnhirspak	tabcnt4.f90
targets.data	Area definitions for diagnostic stats collection	bx	
topo_eqarea_0p1.nc	TOPO product 0.1°	aux, b4	topo2nc.f90
topo_eqarea_0p25.nc	TOPO product 0.25°		topo2nc.f90
topo_eqarea_0p5.nc	TOPO product 0.5°		topo2nc.f90
topo_eqarea_1p0.nc	TOPO product 1.0°	aux	topo2nc.f90
topo_eqarea_2p0.nc	TOPO product 2.0°		topo2nc.f90
topo_eqarea_2p5.nc	TOPO product 2.5°		topo2nc.f90

5 Satellite history sub-system

This section contains the basic descriptions of the modules and data files which make up the Satellite History (SATHIST) sub-system, including PTRANS and REFGE0.

5.1 Overview

From the beginning of the project in 1983 to the present time, there have been many different satellites and centers contributing. The specification of which satellites are to be processed in which positions at which times is defined and controlled by the SATHIST sub-system. SATHIST has five parts, all of which are located in the `prd/lib` directory. The five parts are:

Sat.pm – Individual satellite characteristics such as names, number of channels, etc.

Satpos.pm – Satellite position definitions (sectors)

Sathist.pm – Interface for production system to look-up satellite info for given date

Satevents.txt – Table of satellite events such as start/stop dates, moves, problems, etc.

Spc.pm – Sector processing center definitions

In addition to these SATHIST modules, each satellite has an associated PTRANS file for ozone transmission tables, and geo-stationary satellites have associated REFGE0 files for mapping.

When adding a satellite or center to the system, these components will need to be modified. Each is discussed in more detail in the following sections. The different components work together to uniquely identify the following:

Satellite name (satname) = a single instrument with a particular calibration

Satellite position (satpos) = a processing stream representing a sector of the globe

SPC = a data provider, indicating a particular data format

This means that a particular satellite may be assigned to different positions and/or SPC over time, as its location and data processing format may change. More or less “permanent” changes of assignment are handled by making additional entries in `Sat.pm`. When changes are frequent, such as the seasonal drifting of GOES satellites, the change of location is handled by making keyword entries in `satevents.txt` which apply only to certain months and override the “default” that is set in `Sat.pm` (eg. `SUBSATLON` and `SUBSTITUTE_SATKEY`). Regardless of the assignment of `satpos` or `SPC`, the calibration for a particular satellite is the same and it will use the same HBT calibration tables.

5.2 Satellite definition module (Sat.pm)

The satellite definitions are stored in a Perl hash structure within the `Sat.pm` Perl module.

For each satellite/position/date-range, there is a fixed entry in this table, with a unique sequential hash key identifier, which defines the following:

id: 4-character satellite ID (sssn) assigned by ISCCP which must be unique and consists of a 3-char satellite abbreviation with a 1-char satellite number (1-9,a-z). This is generally called “satid” throughout the code and is the index cross-reference to the Satevents.txt file.

num: Integer satellite number assigned by ISCCP to uniquely identify the satellite. At the beginning of the project in 1983, the following ranges were assigned to each position. Throughout the years, attempts have been made to stay within these ranges but at some point it may not be possible to keep this organization. That should not matter, as long the number is always unique.

10's = NOA

20's = GOW

30's = GOE

40's = MET

50's = GMS

60's = NOM

70's = INS

80's = CMA (china, not currently used)

90's = INP (brazil, not currently used)

name: Full satellite name

datebeg: yyyyymmdd beginning date for the entry (usually the first date data is available). This is not necessarily the first date of operations, rather it is the first date the data is to be used in ISCCP. Date ranges for satellites assigned to the same SPC must not overlap.

dateend: yyyyymmdd ending date for the entry (usually the last date data is available) or 99999999 to indicate data collection is currently ongoing. This is not necessarily the last date of operations, rather it is the last date the data is to be used in ISCCP. Date ranges for satellites assigned to the same SPC must not overlap.

spc: Sector processing center (ie. provider of data to ISCCP). This is the index cross-reference to the Spc.pm table

pos: Satellite position in ISCCP to which this satellite is assigned. This is the index cross-reference to the Satpos.pm table

subsatlon: Sub-satellite longitude for geo-stationary satellites, N/A for polar orbiters.

nchans_b3: Number of channels in B3 data (corresponds to number of channels in BT cal tables, from which HBT was made)

radius_b3: For B3, radius for mapping equations, or 0 if not applicable

refgeo_b3: For B3, name of refgeo file for mapping, or N/A

id_b1: Satellite id in NCDC's B1 filename convention format (B1U or GAC)

spc_b1: Spc in NCDC's B1 filename convention format (B1U or GAC)

nchans_b1: Number of channels in B1 data (B1U or GAC)

radius_b1: For B1, radius for mapping equations, or 0 if not applicable

refgeo_b1: For B1, name of refgeo file for mapping, or N/A

refgeo_flip: Whether to flip the refgeo file horizontally or vertically (H or V or N/A)

chans_b1: Array of hashes, one for each B1 channel in ISCCP order, each has following properties:

name: B1 channel name

bits: Number of bits (8 or 10)

invert: Y or N, whether to invert the IR counts or not

input_chan: Input B1U/GAC channel number (often different from ISCCP order)

cnvtab: Physical conversion table to apply for B4 encoding (RFL or TMP)

band: Bandwidth

solcon: Solar constant for VIS channel

5.2.1 Sample entry

An example of an entry in the perl hash in Sat.pm:

```
1 => { id=>'NOA7', num=>'11', name=>'NOAA-7', datebeg=>'19810801', dateend=>'19850131',
      spc=>'NOA', pos=>'NOA', subsatlon=>'N/A', nchans_b3=>'5', radius_b3=>'265', refgeo_b3=>'N/A',
      id_b1=>'NC', spc_b1,'NOA', nchans_b1=>'5', radius_b1=>'795', refgeo_b1=>'N/A', refgeo_flip=>'N/A',
      chans_b1=>[ { name=>'CH1', bits=>'10', invert=>'N', input_chan=>'1', cnvtab=>'RFL', band=>'0.6' },
                  { name=>'CH4', bits=>'10', invert=>'N', input_chan=>'4', cnvtab=>'TMP', band=>'10.8' },
                  { name=>'CH2', bits=>'10', invert=>'N', input_chan=>'2', cnvtab=>'RFL', band=>'0.9' },
                  { name=>'CH3', bits=>'10', invert=>'N', input_chan=>'3', cnvtab=>'TMP', band=>'3.7' },
                  { name=>'CH5', bits=>'10', invert=>'N', input_chan=>'5', cnvtab=>'TMP', band=>'12.0' } ],
      solcon => '56.688', oldsol => '56.66' }
```

In the above example, “1” at the left is the hash key, and to the right is the hash value as a nested structure. The script sat_print.pl may be used to generate a text file which lists the contents of Sat.pm in a more readable format, and sorted by satellite position. As an example, after running sat_print.pl, in the output file Sat.txt would be this:

```
Pos Num Id      (Id_B1)      Name      Lon Spc (Spc_B1) Datebeg Dateend
NOA  11 NOA7      (NC)        NOAA-7    N/A NOA (NOA)  19810801 19850131 (Key 1)

B3 nchans:  5, Radius: 265, Refgeo:          N/A, Refgeo_Flip: N/A
B1 nchans:  5, Radius: 795, Refgeo:          N/A
ISCCPchan  Name Band Bits Invert InChn Cnvtab
1          CH1  0.6  10   N      1    RFL
2          CH4 10.8  10   N      4    TMP
3          CH2  0.9  10   N      2    RFL
4          CH3  3.7  10   N      3    TMP
5          CH5 12.0  10   N      5    TMP
```

5.2.2 Adding a new satellite

To add a new satellite to the system, first edit the Sat.pm module (step 1 below), then add PTRANS (step 2 below) and REFGeo files (step 3 below) to prd/tables, and finally add the VISCOR processing if necessary (step 4 below). Note that after adding the satellite to the system, before it can actually be used in production it must be calibrated, initial HBT files must be generated, and B1QC must be run. The initial calibration setup and initial HBT generation process is described separately (See Section 7).

Details for the steps to add a satellite to the system are as follows (fields are defined in Section 5.2 above):

1. Edit the Sat.pm module to add an entry to the hash structure, at the next sequential hash key number. Always add new entries at the end. Hash key values already assigned should not be changed. Fill in all of the fields as follows (refer to field descriptions above):

id: Choose a unique 4-character identifier (sssn)

num: Assign a unique number for the satellite, following previous patterns

datebeg: Enter the date to begin processing this satellite

dateend: Enter the date to end processing this satellite

spc: Assign the SPC providing this satellite data (see Section 5.3 to add an SPC)

pos: Assign the satpos for this satellite (see Section 5.4)

subsatlon: Specify the sub-satellite longitude of the satellite (or N/A for orbiters)

nchans_b3: Specify the number of channels in B3 data (ie. number chans in HBT)

radius_b3: Specify the radius for B3 mapping (or 0, see Section 5.9)

refgeo_b3: Specify the name of the refgeo file for B3 (or N/A) and place the refgeo file in prd/tables (see Section 5.8)

id_b1: Specify the satellite ID as it appears in B1U/GAC filenames

spc_b1: Specify the SPC as it appears in B1U/GAC filenames

nchans_b1: Specify the number of channels in B1U/GAC data

radius_b1: Specify the radius for B1U mapping (or 0, see Section 5.9)

refgeo_b1: Specify the name of the refgeo file for B1U (or N/A) and place the file in prd/tables (see Section 5.8)

refgeo_flip: Specify whether to flip the refgeo coordinates if necessary

chans_b1: Fill in the hash array with one line for each B1U/GAC channel, in ISCCP order, with the following fields on each line:

name: Enter the name of the channel as it appears in the B1U/GAC file

bits: Enter the number of bits in a count value (8 or 10)

invert: Specify whether IR counts need to be inverted or not

input_chan: Specify the input channel number order

cnvtab: Specify the conversion table for encoding the channel

band: Specify the bandwidth of the channel

solcon: Enter the solar constant for the VIS channel (see Section 7.2.2)

2. Generate a PTRANS file using the instrument visible (ISCCP channel 1) response function and place it in prd/tables/ptrans (see Section 5.7)

3. If the satellite is pre-mapped by the SPC, place the refgeo file in prd/tables, with name as specified in Sat.pm above (see Section 5.8)
4. If the satellite has a broad VIS channel, where “broad” is defined to be more than 0.3 microns at 50% response points in the spectral response function, it must be added to the VISCOR processing step (See Section 4.7.2). This change might also be needed if the satellite has a spectral response peak at a much longer wavelength than the other channels, say at > 0.7 microns. To add a satellite to the VISCOR processing step, modify the find_viscor and do_viscor functions in Sat.pm as needed to match the satellite ID.
5. Setup the initial calibration files and create initial (V0) HBT files (See Section 7).

5.2.3 Changing the location of a satellite, or its SPC assignment

There are times when a satellite may be re-located to a new position and/or may be processed by a different SPC. This is handled by making an additional entry in Sat.pm for the satellite and adjusting the datebeg and dateend fields so that production switches from one entry to the other on a particular date. The entry for the new location will have the new sub-satellite location and may be assigned to a different satpos and/or SPC, and may have a different refgeo file. All other fields will usually remain the same. New entries should be assigned to a hash key value that is the next sequential number after the largest value used so far. Existing hash key values should not be changed, as they may have been referred to elsewhere, such as in the Satevents file substitute satellite keyword. Use the sat_print.pl script to print the Sat.pm in a more readable form (ordered by satpos).

5.3 Satellite processing centers (Spc.pm)

Over time, various institutions have contributed to the ISCCP project by providing B1/B2 satellite data. These institutions are called “Satellite Processing Centers” or SPC. The Spc.pm Perl module contains a table which assigns a 3-letter acronym to each center. When a new center participates in the project, a new entry is added to this table. The SPC acronym will appear in the product filenames. (Historical note: in early days SPC was defined as “Sector Processing Center” but over time the “S” evolved to “Satellite”)

5.3.1 Adding a new SPC

To add a new SPC to the system, edit the Spc.pm module as follows:

1. Choose a unique 3-character SPC name
2. Assign the next sequential id number
3. Add a new entry to the perl hash following the pattern of previous entries, which specifies the full text of the center’s name. Ignore the num_hdr field as that referred to header files on B2 tapes and is obsolete.

5.4 Satellite positions (Satpos.pm)

ISCCP divides the globe into 7 satellite positions. These are:

1 GMS: Japan sector geosat

- 2 MET: Meteosat sector geosat
- 3 GOW: Goes-west sector geosat
- 4 GOE: Goes-east sector geosat
- 5 INS: Indian ocean sector geosat
- 6 NOA: NOAA afternoon orbiter
- 7 NOM: NOAA morning orbiter

The order 1-7 shown above is the index of each position in the data arrays which form the satellite hierarchy, that are produced by the SATHIER step. This is not the same as geographic order. The geographic order is GOW, GOE, MET, INS, GMS, NOA, NOM.

Over time there may be different satellites assigned to these sectors in the Sat.pm module. The sectors represent fixed positions in the satellite hierarchy for merging to the global grid. These satpos designations are used to identify the different satellite processing streams in production, while the actual sub-satellite longitude of the satellite assigned to each position is used to map the data and to generate the hierarchy for merging. Historically there have been 7 positions, however it is possible to define additional positions. Depending on data availability, some positions may not have any satellites assigned to them for a given processing month, and in that case there is no processing stream for that position.

5.4.1 Adding a new satpos

Should there be a need in future to add satellite positions to the hierarchy, this can be accomplished by the following steps:

1. Choose a 3-char identifier for the new position and assign it a code number (eg. 8)
2. Add the identifier and code number to the CODES list in Satpos.pm
3. Add a flag indicating satellite type in the ISPOLAR list in Satpos.pm
4. If the new position is a polar orbiter type, add entries in IOFF and IOFF_B1U lists in Satpos.pm to indicate mapping offsets
5. Add the new position to SATTAB in d1prod.f90 (currently there are placeholders of "000" for potential positions 8 through 12)
6. Add the new position to the SATPOS list in sathier_plot.pro in order to plot the new hierarchy properly
7. Modify both d1prod.f90 and d2calcor.f90 to increase the dimension of the pairs array (ie 8x8 instead of 7x7) this is the array which holds the overlap statistics used to compute the calibration corrections in CALCOR
8. Modify the environment variables ISCCP_SATPOS_N and ISCCP_SATPOS_LIST in the prdenv script so that the new position will be processed by prdrun
9. Add a new sub-directory for the satpos to the prd/data directory and stage the data there

5.5 Satellite history events (Satevents.txt)

The satevents.txt file is a text file which is maintained by hand using any text editor. The purpose of this file is to record any known events relevant to a satellite which may affect the processing results. These events may include motion (change of sub-satellite point), calibration events, start/stop operations dates/times, QC notes, etc. When production is scheduled, Sathist.pm searches for entries that are within the date range of the scheduled production, and any entries

found are displayed on the schedule GUI. This is information for the operator, things to remember and consider when evaluating the results. In addition, there are certain keywords which will be used by the system automatically, to adjust the sub-satellite point for moving satellites, and to identify a substitute satellite.

5.5.1 SUBSATLON keyword

There are times when satellites may be moving throughout the month, for example when migrating to a new position. In this case, an average location is computed by hand, and entered into the Satevents.txt file with the SUBSATLON keyword. The number that follows the keyword is automatically used as the sub-satellite longitude (degrees east) instead of the default that was set in the satellite definition (Sat.pm).

5.5.2 SUBSTITUTE_SATKEY keyword

There are times where more than one satellite is available for use in a given position. If data is missing from the primary satellite, it is often desirable to substitute data from an alternate satellite with the same view. If this is to be done, the Sat.pm entry defines the primary satellite for the date range. The alternate satellite is specified via the SUBSTITUTE_SATKEY keyword. The number following the keyword specifies the alternate hash key number to use for looking up the substitute satellite specifications. This is used by b4prod to search for substitute images when primary satellite images are not found. Once ingested, the substitute images are labeled as primary satellite for downstream products. The keyword is tagged with a date in the middle of the month, and will apply to the whole calendar month being processed.

5.5.3 Entry format

Each entry is one line with three fields separated by colons, as follows:

sssn:yyyymmdd:event

sssn is the 4-character satellite ID defined in the Sat.pm module

yyyymmdd is the year, month and day of the event

event is whatever text you want to describe the event, or a keyword followed by a value

Examples of valid entries:

```
GOE6:19861120:BEGAN DRIFTING WEST AT 0.46 DEG/DAY      (just informational)
GOE6:19861201:SUBSATLON -105.53 (MONTHLY AVERAGE)    (prd uses -105.53 as SSP)
MET4:19900115:SUBSTITUTE_SATKEY 20 MET 3              (prd looks for MET3 if MET4 n/a)
MET3:19930419:FROM SLOT 19 - DECONTAM. OF IR/WV USE ONLY VIS IMGS (addl QC info)
```

5.6 Sathist lookup function (Sathist.pm)

Each production step has an associated run script which queries Sathist to get all of the satellite-specific information needed. This is done via the sathist() function in Sathist.pm. Sathist looks up the specific satellite info for a given date range and satellite position and searches for events within the date range, and returns this information to the script. The first time sathist is called for a given satellite position, it writes some information to a small text file in the satpos sub-

directory of the wrkdir, called yyyy_mm.hist. In addition, the information is displayed in the prdsched GUI, including the events, for review by the operator.

5.7 PTRANS tables

Each satellite has an associated PTRANS table which contains ozone transmission tables. These tables are computed manually when a new satellite is added to the system, by running a program which inputs a static table of absorption vs. wavelength and the response function of the instrument visible channel (ISCCP channel 1) and outputs the PTRANS table for the satellite. These tables are placed in the prd/tables/ptrans directory, and are used by the dxprod step. PTRANS files are supplied for all satellites processed so far, and will need to be generated for future satellites.

New PTRANS files for future satellites may be generated with the IDL program `mk_ptrans4newsat.pro` which resides in `prd/src/sat_setup/ptrans` directory. This program uses several string-handling functions from the “`idlastro`” public IDL library which are also supplied.

The inputs to the program are located in the `prd/tables/ptrans/input` directory. The input files are:

<code>linefile_only_O3_0.bin</code>	Absorption optical depths for 50 values of ozone 0.1 to 5.0 cm-STP
<code>max_min_mmin_spectra.txt</code>	For solar irradiance in nW/m2/nm
<code>visresponse.sssnn.txt</code>	Response function for satellite “ <code>sssnn</code> ” VIS channel, <code>noa09</code> supplied as example. Create this manually, by copying the VIS channel information from the all-channel response function table in the <code>calibration/sat_response_functions</code> directory. Only the first and last entry should have 0.0 in the first column (response values). Remove all other leading/trailing zeroes. The program will interpolate over second column (wavelength) between these entries.

To run, build the IDL program and call the `isccp_trans` function with the name of the response function file for the satellite being processed. For example `noa09`:

```
$ cd prd/src/sat_setup/ptrans
$ idl
IDL> .r mk_ptrans4newsat.pro
IDL> isccp_trans, 'visresponse.noa09.txt'
```

This reads `linefile_only_O3_0.bin` and calculates the transmission through ozone for the response function for 50 ozone values. The output is written to the file `ptrans_sssnn.data` (eg. `ptrans_noa09.data`). Two postscript plot files are also generated: `ptrans_sssnn.data.ps` and `visresponse.sssnn.txt_ref_plot.ps`. These plots serve as a visual check of the results. If the plots look good, then move `ptrans_sssnn.data` to the `prd/tables/ptrans` directory for use in production.

5.8 REFGEO tables

For satellites which are already mapped in the Normalized Geostationary Projection (see Tech Pub from Eumetsat at www.cgms-info.org) a REFGEO file is used which contains the latitude and longitude for the pre-mapped satellite data (typically Meteosat, INSAT and MTS, and newer satellites). These files are placed in prd/tables. The filename (it can be named anything you want) is specified in the Sat.pm module entry for the satellite. If the coordinates need to be flipped horizontally or vertically, set the appropriate flag in Sat.pm. These files are used in the AUX step. For geo-stationary satellites which are not pre-mapped, the mapping is determined according to the radius values specified in the Sat.pm module (see Section 5.9).

The software to create REFGEO files is obtained from the satellite operator. There are programs available for MET, MTS and MSG satellites. These codes reside in prd/src/tables/refgeo_b1 directory, and were used to create the REFGEO files in prd/src/tables. These are:

Program name:	File name:	Used for:
create_refgeo_met_b1.f90	refgeoB1.met	MET-2,3,4,5,6,7
create_refgeo_msg_b1.f	refgeoB1.msg	MET-8,9 (MSG-1,2)
create_refgeo_mts_b1.f	refgeoB1.mts	MTS-1,2

Note that for MET satellites, the sub-satellite longitude is automatically added to the REFGEO locations in auxgeo.f during production, so the same refgeoB1.met file can be used for MET at any location.

To use REFGEO files for new satellites:

1. Identify appropriate existing REFGEO filename, or create new REFGEO file using software obtained from satellite operator if necessary
2. Edit Sat.pm to add the name of the REFGEO file to the entry for the satellite
3. Edit Sat.pm to specify the flip direction, if needed (H or V) to match the images
4. Check auxgeo.f90 to make sure the satid will be recognized in the “if” statement for REFGEO (beginning on line 328, in the “mapper” subroutine), and that the format will be read properly. If necessary, add another clause to the if-statement to match an additional satid.
5. Check spacetest_module.f90 to make sure the satid will be recognized in the “if” statement for coordinate mapping (beginning on line 289, in the “space_test” subroutine), and that the flipping is being done properly. If necessary, add another clause to the if-statement to match an additional satid.

5.9 Radius

For **geostationary** satellites which are not pre-mapped and therefore do not use REFGEO files (see Section 5.8) the radius for the mapping equations must be specified in Sat.pm. This number is chosen to minimize the loss of data when mapped using General Perspective Projection equations (see MapProjections_GeneralPerspectiveProjection.pdf) which is used by geomap.f90 in the aux and b4prod steps to map the data, and represents the earth radius in pixels for the image. An IDL routine has been created which iterates through various values of radius to find

the optimum value for each satellite. The optimum value is the value for which the number of lost pixels is approximately 3% of the total. This balances minimal loss of data with minimal empty cells in the resulting map. This code resides in prd/src/tables/radius directory.

To compute the radius for geosats:

1. Extract the lat, lon and IR data from one B1U file into `fltarr(3,nx,ny)` and write that to a binary file using IDL. See `README.radius.ncdc` in the `prd/src/tables/radius` directory for the suggested format of this file.
2. Feed that file into the IDL routine (modify the read statements for your extracted file format if necessary), with a starting value for the iteration. Optionally, `FLAG=1` can be set inside the program to produce plots of remapped IR image. The default is `FLAG=0`.

```
IDL> .r geo_remap_radius_4ncdc.pro
IDL> b4geo_b1k_map4ncdc, extracted_data_filename, 700L
```

3. Look at the last line of the output file (`extracted_data_filename.txt`) to see the optimal radius printed there. It is the value at which the number of pixels lost is 3%.
4. Edit `Sat.pm` to add that radius value to the entry for the satellite
5. Optionally, plots can be created to show radius vs. loss/empty from the output `.txt` file. The plots will be named `extracted_file_name.txt.ps`.

```
IDL> plot_percent_loss, extracted_file_name.txt
```

For **polar orbiting** satellites the radius for the mapping equations must be specified in `Sat.pm` and the offset for the storage array must be specified in `Satpos.pm`. The radius and offset are used by `polmap.f90` in the `aux` and `b4prod` steps to map the data for the polar caps (S,N sectors) using Lambert Azimuthal Equal Area equations. For GAC data, these numbers have been chosen to generate a map array that is approximately three times the size of the B3 maps. The mid-latitude data is mapped to a square lat,lon map (A,D sectors).

5.10 Initial calibration setup (nominal calibration, HBT V0)

The process for setting up the initial calibration tables for new satellites is described in Section 7. This will need to be done for satellites from 2010 forward.

6 B1U/GAC Quality Control (B1QC)

6.1 Statistics computed

The `b1uqc.f90` program is run manually as needed. This process requires HBT with at least nominal calibration (see Section 7). The QC program reads calibrated B1U or GAC images for a month and calculates for the VIS and IR channel 23 statistics (also for `mue`, `mue0` and `subsat`, `subsol`). Radiances must be calibrated (at least nominal tables must be available) before running B1QC. The percentile tests are applied to calibrated counts. The statistics calculated are:

- 1) pixel & scanline counts
- 2) 10th percentile
- 3) 25th percentile
- 4) histo. medians
- 5) 75th percentile
- 6) 90th percentile
- 7) image entropies*
- 8) full img. means
- 9) navigation hi's
- 10) auto-correl. coef. means
- 11) auto-correl. coef. stdev
- 12) auto-correl. coef. dif means
- 13) auto-correl. coef. dif stdev
- 14) vis-ir crscor
- 15) scan line total mean of means
- 16) scan line total stdev of means
- 17) scan line total mean of mean dif
- 18) scan line total stdev of mean dif
- 19) col. totals mean of means
- 20) col. totals stdev of means
- 21) col. totals mean of mean dif
- 22) col. totals stdev of mean dif
- 23) median scanline

*The entropy is calculated from a histogram (8-bit precision) of radiances from the whole image using $H = -\sum_0^{255} P_i \log_2 P_i$, where P_i are the frequencies normalized so that their sum equals 1.0.

The results are written to the output file (`BQC.satpos.satid.yyyymm`):

- array `stats(31days,8+1 or 24+1 gmts,8 panels,23 statistics)`
- array `ovr(31days,8+1 or 24+1 gmts,8 panels,23 statistics)` that stores info about how far each image stat is from the gmt monthly mean in standard deviation. This is used to flag images that are more than 3 standard deviations from the monthly gmt mean. These images will be plotted red on the plot and also listed in `qcresults_BQC.satpos.satid.yyyymm.txt`.
- array `mn(8+1 or 24+1 gmts,8 panels,23 statistics)` gmt monthly means
- array `sig(8+1 or 24+1 gmts,8 panels,23 statistics)` stdev
- array `ngmt(31,8)` number of gmts

The extra index in the gmt dimension (8+1 or 24+1) is used to store the daily mean.

The results are plotted as deviations from monthly averages of the above quantities and large deviations (more than 3 standard deviations away) are flagged with colored symbols.

6.2 Running B1QC

The B1QC process is run manually as needed.

1. Set the regular production environment variables by sourcing the prdenv script
2. In the working directory run the b1uqc.run script for each satellite:
b1uqc.run <working_directory> year month ddbeg ddend satpos
eg: b1uqc.run . 2007 07 01 31 met

The output files will be:

BQC.met.MET-9.200707 -> the stats for one month

b1uqc.met.out -> the printed output from b1uqc.f90

6.3 Plotting results (qcresults)

To create the QC plot and generate the QCRESULTS textfiles, use IDL:

```
IDL> b1uqc_plot, 'BQC.met.MET-9.200707'
```

This will create the plot: BQC.met.MET-9.200707stat.ps

and write the text file: qcresults_BQC.met.MET-9.200707.txt

The qcresults_* file has at the end the monthly mean for some of the 23 stats to monitor the change from month to month.

NOTE: For geostationary satellites, the qcresults file column “gmt” indicates the nominal timeslot of the image 00,03,06,09,12,15,18,21. For polar orbiter satellites the column “gmt” is an index 1-24 because there can be 1-3 files (usually 2) whose start time falls within the 3-hour window at each timeslot (timeslot window = +/- 1.5 hours around the nominal GMT). The correspondence between this index and the actual orbit file name can be found by looking into the log from the program run. The index “igmt” is printed as each file is processed.

For example, one day of polar orbiter data may be assigned “igmt” index values as follows:

The GMT 0 timeslot includes any image that has the start time ‘S’hhmm between S2230 (previous day) and S0129 (current day) => time interval [S2230,S0130)

IGMT index 1: NSS.GHRR.NM.D07121.S2254.E0021.B2521415.GC

IGMT index 2: NSS.GHRR.NM.D07122.S0015.E0144.B2521516.WI

IGMT index 3: no orbit

For GMT 3 interval [S0130,S0430)

IGMT index 4: NSS.GHRR.NM.D07122.S0139.E0324.B2521517.WI

IGMT index 5: NSS.GHRR.NM.D07122.S0319.E0514.B2521618.GC

IGMT index 6: no orbit

For GMT 6 interval [S0430,S0730)

IGMT index 7: NSS.GHRR.NM.D07122.S0508.E0654.B2521819.GC

IGMT index 8: NSS.GHRR.NM.D07122.S0649.E0835.B2521920.GC

IGMT index 9: no orbit

For GMT 9 interval [S0730,S1030)

IGMT index 10: NSS.GHRR.NM.D07122.S0830.E1025.B2522021.WI

IGMT index 11: NSS.GHRR.NM.D07122.S1019.E1214.B2522122.WI

IGMT index 12: no orbit

For GMT 12 interval [S1030,S1330)

IGMT index 13: NSS.GHRR.NM.D07122.S1209.E1402.B2522223.WI

IGMT index 14: no orbit

IGMT index 15: no orbit

For GMT 15 interval [S1330,S1630)

IGMT index 16: NSS.GHRR.NM.D07122.S1357.E1541.B2522324.WI

IGMT index 17: NSS.GHRR.NM.D07122.S1536.E1730.B2522425.GC

IGMT index 18: no orbit

For GMT 18 interval [S1630,S1930)

IGMT index 19: NSS.GHRR.NM.D07122.S1725.E1858.B2522526.GC

IGMT index 20: NSS.GHRR.NM.D07122.S1852.E2037.B2522627.GC

IGMT index 21: no orbit

For GMT 21 interval [S1930,S2230)

IGMT index 22: NSS.GHRR.NM.D07122.S2034.E2217.B2522728.GC

IGMT index 23: NSS.GHRR.NM.D07122.S2212.E2357.B2522829.GC

IGMT index 24: no orbit

For GMT 0 next day

IGMT index 1: NSS.GHRR.NM.D07122.S2352.E0123.B2522930.WI

The key is to remember that for each time slot (interval) the files are in increasing start time order and this is how to fill the index:

Timeslot GMT 0 = IGMT index 1,2,3	Interval [S2230,S0130)
Timeslot GMT 3 = IGMT index 4,5,6	Interval [S0130,S0430)
Timeslot GMT 6 = IGMT index 7,8,9	Interval [S0430,S0730)
Timeslot GMT 9 = IGMT index 10,11,12	Interval [S0730,S1030)
Timeslot GMT 12 = IGMT index 13,14,15	Interval [S1030,S1330)
Timeslot GMT 15 = IGMT index 19,20,21	Interval [S1630,S1930)
Timeslot GMT 21 = IGMT index 22,23,24.	Interval [S1930,S2230)

6.4 Plot interpretation

All of these plots have dynamic vertical scales. In these plots GREEN dots indicate values off-scale low, BLUE dots indicate values off-scale high and RED dots indicate values flagged as out of range.

Page 1: “PIX & SCLN CNTS” provides for all individual images in a month the number of Water Pixels, Land Pixels and Coast Pixels, the number of scan lines, the number of on-planet pixels, day pixels and night pixels. These counters are much more variable for polar orbiter data than for geostationary data; the former should not exhibit any changes of pattern over the month and the latter should exhibit near stability except for day and night pixel numbers, which should vary diurnally. The exception is truncated images which may exhibit more variation.

Pages 2-6: Provide radiance distribution information as 10th, 25th, median, 75th and 90th percentile Count Values from whole-image histograms for Channel 1 (VIS) and Channel 2 (IR) separated for water and land areas. Lower percentile values should be less variable especially over water, but there should be some variation with no sudden changes in level. Polar orbiter statistics are generally more variable than geostationary statistics and depend on changes of orbit swath length.

Page 7: “IMAGE ENTROPIES” are a measure of the within-image radiance variability and should exhibit similar patterns of variation from image to image over the whole month.

Page 8: “FULL IMG MEANS” shows the image-mean radiances in Counts for Channel 1 (VIS) and Channel 2 (IR) followed by these values in physical units, percent scaled radiance for VIS and brightness temperature in Kelvins for IR. These statistics are more variable for polar orbiter than geostationary images.

Page 9: “NAVIGATION HI'S” provide image-by-image information about the sub-satellite latitude and longitude and the sub-solar latitude and longitude. What is actually presented is the latitude and longitude of the largest values of the cosine satellite and solar zenith angles found within each image. For polar orbiter data, the sub-satellite information should exhibit a nearly regular pattern of variation over the month. For geostationary data, the sub-satellite latitude and longitude should be nearly constant over the month but may undergo a small diurnal oscillation. Sudden changes in the sub-satellite behavior for geostationary satellites indicate orbit trim maneuvers. The sub-solar latitude should vary only very slowly over the month, whereas sub-solar longitude should vary depending on the time of day sampling of the satellite type: for geostationary satellites there should be a regular diurnal variation over a range of about 100-150° but for polar orbiters the diurnal range is 360°.

Page 10-13: Provide the average, the standard deviation, the average difference and the standard deviation of the differences over whole images of the along-scanline, pixel-to-pixel, autocorrelations of the cosine of the satellite (“mu”) and solar (“muo”) zenith angles and the Channel 1 (VIS) and Channel 2 (IR) radiances. There should be very little variation in these quantities.

Page 14: “VIS-IR CRSCOR” provides the average, the standard deviation, the average difference and the standard deviation of the differences over whole images of the pixel-level cross-correlation of the two radiance images. Although the average and standard deviation values vary a little over the diurnal cycle because of sampling changes, the other two quantities should not vary.

Pages 15-18: Provides the average, the standard deviation, the average difference and the standard deviation of the differences over whole images of the scanline radiance means, standard deviations, mean and standard deviation of scanline-to-scanline radiance differences. There is notable variation of the scanline statistics, though the magnitude over the month should be similar image to image, but the scanline difference statistics should exhibit much smaller variations over the month.

Pages 19-22: Provides the average, the standard deviation, the average difference and the standard deviation of the differences over whole images of the “image column” radiance means, standard deviations, mean and standard deviation of “column-to-column” radiance differences. There is notable variation of the column statistics, though the magnitude over the month should be similar image to image, but the column difference statistics should exhibit much smaller variations over the month.

Page 23: This plot shows the scan line number where we find the median value for the number of pixels. Example: if the total number of pixels is 444444 the pixel number 222222 is in scan line 1230 and this number is on the plot.

6.5 Visual inspections

Images flagged in the qcresults file are to be visually inspected to see whether they should be dropped. The two channels being inspected are Infrared (IR) and Visible (VIS). The IR channel measures temperature and the VIS channel measures brightness. “Counts” are the values each pixel is assigned by the satellite. We then assign a value to each pixel based on calibration information. Count values range from 0 to 255. Value 255 represents null or undefined (eg. off-planet pixels). Scan lines are horizontal lines of pixels read sequentially by the satellite instrument. Each image is made up of a finite number of scan lines.

Each image inspected is assigned a QC error code from Table 6-1. The QC error code is entered under the appropriate column in the QCRESULTS file. A channel code is used to indicate an image that is not to be used in any further processing. The presence of a channel code signifies a FATAL error. When using a channel code you are indicating the channel in which the problem occurs. To verify that this particular image was meant to be deleted from production (aka “quarantined”), indicate the corresponding QC error code to describe the situation.

Table 6-1 Definition of QC error codes

Error Code	Definition	Status
1	1 Hour Late	Not fatal

	<p>The difference between the time given for the image in the QCRESULTS file and what is seen by the inspector is greater than one hour and less than two hours. If the difference is two hours or more then use error code 2.</p>	
2	<p>2 Hours Late</p> <p>The difference between the time given for the image in the QCRESULTS file and what is seen by the inspector is greater than two hours. Use this error code if a different GMT is seen from what is indicated in the QCRESULTS file.</p>	FATAL
3	<p>Bad Line Segments</p> <p>This error is used to indicate that there is at least one scan line which is made up of part good pixels and part bad. This would look like the pattern represented in the scan line had a bad sequence of pixels in it but the rest of the scan line and image has recovered the pattern.</p> <p>If the number of these bad segments make up more than 10% of the image, then use error code 5, which is fatal.</p>	Not fatal
4	<p>Less Than 10% Bad Scan Lines</p> <p>The image contains scan lines that are completely bad (unlike bad segments), but the number of these make up less than 10% of the entire image. It doesn't matter what the bad scan lines consist of, i.e. VIS in the IR channel, just as long as these scan lines make up less than 10% of the image.</p>	Not fatal
5	<p>Greater Than 10% Bad Scan Lines</p> <p>The image contains scan lines that are completely or even partially bad AND make up more than 10% of the image.</p>	FATAL
6	<p>High Entropy</p> <p>The image contains an extreme variation in counts (i.e. it looks noisy). Use this code only if the noise appears for LESS THAN 10% of the image. Otherwise, use this code in conjunction with code 5 which will kill the image from further processing.</p>	Not fatal, unless also with code 5.
7	<p>Low Entropy</p>	Not fatal, unless with

	<p>The image contains very little variation in counts or there is a piece of the image missing (not 0 or 255). If you can detect this problem easily then there is a problem. Indicate such a problem by specifying the channel code and this error code. The channel code will kill the image from further processing.</p> <p>If the QC plot indicates Low Entropy but you cannot see it then just indicate this code without a channel code making this not fatal.</p>	channel code
8	<p>Dashes</p> <p>The image contains a scan line segment which looks as if the instrument was sticking at a point. An example of this would be something that looks like an artificial cloud or a cloud which has a geometric shape, i.e. a square or a rectangle. Another example of this would be a vertical line through an image. This would indicate that the instrument was sticking every time it arrived at that pixel location.</p>	Not fatal
9	<p>Bad Section</p> <p>The image contains LESS THAN 10% bad scan lines but these bad scan lines are in a sequence. (Please use this error to indicate that the scan lines are in a sequence so that if someone was ambitious in the future, they could reprocess the image omitting this sequence of scan lines). This is also called Jumping Latitude, where a section of the image is missing but when it resumes at higher or lower latitudes the rest of the image is navigated correctly.</p>	Not fatal
10	<p>More than one image in the channel</p> <p>This error code is used to indicate that more than one image appears in the channel (not used to indicate VIS scan lines in the channel – that is considered bad scan lines and handled as such). An example of this would be the beginning of an image which goes on to the equator which is immediately followed by another image</p>	FATAL
11	<p>Navigation Error</p> <p>If the navigation is off on the image: the coast lines do not match up to the image (i.e. the image appears shifted), the coast lines go off planet, there are chunks missing from the image that aren't filled by zero's or 255's, etc.</p>	FATAL

12	MISCELLANEOUS This is used to describe FATAL miscellaneous errors that do not fit into any other defined category. Indicate the channel in which the error appears and give a description of the problem beginning on the first line in the QCRESULTS file to verify that you want this image dropped from further processing. If you want to make a comment on an image but do not want to kill it, then use zeros in the channel and error code columns and then write whatever comment you want following.	FATAL
13	Extremely HOT/COLD Indicating that the image needs to be dropped from further processing because of extreme temperature. This also puts out a FLAG for the person in charge of calibration to inspect this image after recalibration. (Remember – The decision to kill the image should be considered final).	FATAL
14	Compressed Polar Orbiter Images This error code can only be used for polar orbiter images. The image contains a number of missing scan lines distributed among the image, but the navigation is correct. The effect is that the image looks correct but compressed (squashed).	Not fatal

7 Initial Calibration Setup

This section describes the calibration setup process for new satellites including nominal calibration and initial HBT creation.

7.1 Overview

The ISCCP calibration tables are contained in the HBT files. Each satellite has three calibration tables for the IR and VIS channels: nominal (NOM), normalized (NORM) and absolute (ABS). The initial setup with nominal calibration must be done before running QC. The absolute calibration results are required for production. The normalized calibration is used in generating the absolute coefficients.

7.1.1 Nominal calibration (NOM)

Some satellites provide the NOM calibration information as a direct count-to-radiance conversion table. The NOM calibration coefficients come from the SPC providing the radiance data and are based on pre-launch, on-board or other estimates of the calibration.

7.1.2 Normalized calibration (NORM)

The NORM calibration is different for the polar orbiters and geostationary satellites. For the polar orbiters, the normalization refers to the results of a MONITOR process that compares the distributions of both channel radiances to the distributions from the reference standard, which is the NOAA-9 AVHRR prior to 2010 and the NOAA-18 AVHRR afterwards. For the geostationary satellites, the normalization prior to 2014 refers to the results of comparing the radiances to collocated and coincident observations from the contemporaneous afternoon polar orbiter AVHRR using its nominal calibration. After 2014, a NORM process is used that compares spatially matched radiance distribution statistics from the geostationary and afternoon polar orbiter over each month. This new NORM process is under development.

7.1.3 Absolute calibration (ABS)

The ABS calibration represents the final best estimate of calibration and is obtained differently for the polar orbiters and geostationary satellites. For the polar orbiters, the ABS coefficients represent the correction for the trend of normalized calibration over the lifetime of the particular AVHRR based on results from the MONITOR process. For the geostationary satellites, the ABS coefficients combine both the NORM and ABS corrections to the contemporaneous AVHRR to adjust the geostationary NORM to the final calibration.

7.2 New satellite setup (NOM)

7.2.1 Response functions

The instrument response function is obtained from the satellite operator, and stored in a text file that is edited by hand. The response function gives the radiometer response as a function of wavelength (microns) for each channel. The response function files are stored in <ISCCP_ROOT>/calibration/sat_response_functions directory. The filenames are response.sss-

nn, and each file contains two columns for each channel, representing response (normalized) and wavelength (microns). The responses must be normalized (divide by the max) so that the peak response is 1.0

Example: response.goe-12 (partial, only the first few lines shown)

```

Resp1  Wavel  Resp2  Wave2  Resp3  Wave3  Resp4  Wave4  Resp5  Wave5
0.0397 0.5110 0.0049 4.1256 0.0444 7.4800 0.0598 11.3340 0.0076 13.8908
0.3032 0.5210 0.0048 4.1171 0.0719 7.4521 0.0825 11.3186 0.0104 13.8715
0.5893 0.5310 0.0047 4.1086 0.1179 7.4245 0.1182 11.3033 0.0113 13.8523
0.7033 0.5410 0.0046 4.1002 0.1930 7.3970 0.1785 11.2880 0.0205 13.8332
0.7923 0.5510 0.0046 4.0918 0.3063 7.3697 0.2643 11.2727 0.0395 13.8141
0.8573 0.5610 0.0050 4.0835 0.4556 7.3427 0.3679 11.2575 0.0665 13.7950
0.8942 0.5710 0.0080 4.0751 0.6265 7.3158 0.4841 11.2423 0.1206 13.7760

```

7.2.2 Solar constant (SOLCON)

The solar constant is computed manually for the VIS channel and entered into Sat.pm. The solar constant is the integral over wavelength of the product of the response function and the solar spectrum. The table of solar spectral irradiance is documented in *Description of Reduced Resolution Radiance Data (WMO/TD-NO.58) page 99*. This table was extended beyond 1.2 microns using *The Neckel and Labs Revised Extraterrestrial Spectrum and Atmospheric Absorption Coefficients at 122 Wavelengths* which is available online at:

<http://rredc.nrel.gov/solar/pubs/spectral/model/tables.html>

7.2.3 Polar orbiter nominal calibration (SATINFO)

The information required for nominal calibration of polar orbiters must be obtained and entered by hand into a text file created with the name: sssnn.satinfo. These satinfo files are then stored in the isccp_root/calibration/nominal directory, and are read by the initial HBT creation process (gachbt).

Example: NOA17.satinfo

```

43.36 76.55 3.99          ! solar const for B1 ch1,2 and 3a
0.026983 0.008092 0.006366 ! bandwidth for B1 ch 3b, 4 and 5
0.055500000 0.1627000000    ! slope1,slope2 B1 ch1 first day of data
0.054300000 0.1621000000    ! slope1,slope2 B1 ch2 first day of data
0.026500000 0.1860000000    ! slope1,slope2 B1 ch3a first day of data
-2.219300 -55.963500        ! int1, int2 B1 ch1 first day of data
-2.122700 -56.216000        ! int1, int2 B1 ch2 first day of data
-1.115300 -81.252000        ! int1, int2 B1 ch3a first day of data
498 500 499                ! intersection points B1 ch1,2 and 3a
0 8 6                      ! a3 coef for B1 IR channels 3b,4,5
2669.3554 926.2947 839.8246 ! centroid number
-1.706855 -0.272011 -0.309483 ! A coef for effect T to equiv blackbody T
1.002629 1.001207 1.000989   ! B " " " "

```

7.2.4 Thermal channel tables (SATTABLE)

The SATTABLE.ALL file contains lookup tables for converting radiances to temperatures, temperatures to radiances, or counts to radiances. The kind of table that is stored differs from SPC to SPC. Each entry in the file is a radiance table group with an 80-character header that specifies the satellite ID, the number of tables for that satellite's radiance group (n), and the beginning and ending dates (yyddd), and the beginning and ending UTC times (hhmmss) that the

radiance group covers. Following this header are the N tables, each contained in a 6000 word record. One satellite can have more than one grouping of tables corresponding to the different time periods covered. For each SPC there was a separate source code that created the entry for the specified satellite and time period and added it to the SATTABLE.ALL file. This file will need to be updated to add appropriate sets of tables for future satellites as needed. New code must be developed to do this using satellite-specific information as it becomes available for new satellites.

Format of an entry:

80-char header satid ntabs begdat enddat begtim endtim

6000 words table 1

6000 words table 2

...

6000 words table n

This repeats for next entry

The SATTABLE.ALL file is handled by `sattable_module.f90`, and is used by `btcalcor`, `b1uhbt` and `gachbt` programs. The `sattable_module` contains a utility function `sattable_scan(lun, filename, endianflag)` which can be called from any Fortran test program to read and print all the headers currently within the file as a convenience. An example of this output is in `prd/src/sat_setup/sattable/sattable_headers.txt`.

Because the number and type of tables varies, the `sattable_module` contains a function which identifies which table is the IR RAD(TEMP) table for use by the `btcalcor`, `b1uhbt` and `gachbt` programs. When new tables are added, this function must be modified as needed.

Historically, the tables were created as follows, channels are in B2 order:

NOA 6 tables 2 per channel rad(temp), temp(rad)

Radiance is calculated for a particular temperature, RAD(TEMP), by doing a sum over the spectral range of the radiometer of the Plank function multiplied by the radiometer response. This table is inverted to get TEMP(RAD) and a linear interpolation is used to fill the gaps in the table.

CSU 2 tables 2 per channel rad(count), rad(temp)

To build the RAD(COUNT) table the relationship between temperature and count as specified by CSU is used and a sum over the spectral range of the radiometer of the Plank function multiplied by the radiometer response is done to calculate RAD(COUNT). The tables are combined to get RAD(TEMP) and a linear interpolation is used to fill the gaps in the table.

ESA, EUM, MTI 4 tables 2 per channel rad(temp), temp(rad)

EUM provides tables for converting temperatures to radiances, RAD(T), for both thermal channels. The tables are included in the calibration documentation and they are updated quarterly. These were stored in the ESARAD dataset. The header record shows the satellite id, and the beginning and ending dates for the tables that follow. Each data

record has radiance values for (IR,WV) pairs corresponding to one temperature value. There are five pairs on each record. The first temperature value is 100K and each temperature value after increases by 1K.

```
SAMPLE ESARAD ENTRY HEADER RECORD
SATIDxxx  YYDDD  YYDDD
METEO-2   83182   83273
```

```
SATIDxxx - satellite id
YYDDD   - beginning date of table
YYDDD   - ending date of table
```

```
SAMPLE ESARAD DATA RECORDS
  IR      WV      IR      WV      IR      WV      IR      WV      IR      WV
0.002    0.000  0.002    0.000  0.002    0.000  0.002    0.000  0.003    0.000
0.003    0.000  0.003    0.000  0.004    0.000  0.004    0.000  0.004    0.000
0.005    0.000  0.005    0.000  0.006    0.000  0.007    0.000  0.007    0.000
0.008    0.000  0.009    0.000  0.010    0.000  0.011    0.000  0.012    0.000
```

The ESARAD data set is used to create two tables for each channel (IR and WV): RAD(TEMP), and TEMP(RAD). The RAD(TEMP) tables are expanded, inverted, and a linear interpolation is done to fill in the gaps.

JMA 1 table 1 per channel rad(temp)

Included in a B2 image header record are two tables, RAD(COUNT) and TEMP(COUNT). The IR channel normalization factor is applied to correct the radiance table. The temperature table is corrected for the clamps at the endpoints at 330K and 170K by extrapolation. Then the two tables are combined to create a RAD(TEMP) table using a linear interpolation to fill in the gaps.

UWS 2 tables 2 per channel rad(count), rad(temp)

To build the RAD(COUNT) table the relationship between temperature and count as specified by CSU is used and a sum over the spectral range of the radiometer of the Plank function multiplied by the radiometer response is done to calculate RAD(COUNT). The tables are combined to get RAD(TEMP) and a linear interpolation is used to fill the gaps in the table.

AES 8 tables 2 tables per channel rad(count), rad(temp)

No description available. Presumably handled the same as CSU/UWS.

NOTES:

1. For GOES-6 and GOES-7 there were special situations due to errors in the data. Therefore, for these satellites there are multiple entries with different sets of tables. There are either 2 per channel (rad(cnt) and rad(tem)) or 3 per channel (old rad(cnt), new rad(cnt) and rad(tmp)).

2. For later satellites with more channels than the above historical definitions, the number of tables is expanded to include tables for all thermal channels. For example, the historical description of JMA was 1 table because early GMS satellites had only one thermal channel. Later GMS-5 has 3 tables, 1 for each of the 3 thermal channels.

3. When reading SATTABLE.ALL, for correct identification of the IR RAD(TEMP) table to use in btcalcor, b1uhbt and gachbt, refer to the “isattab” index that is set in sattable_module.f90. Whenever tables are added, this index setting must be updated in the module.

7.3 Initial coefficient files (NORM, ABS)

The NORM and ABS coefficients are stored in tables that should be kept in a version control system (such as svn) for documentation. Initially the tables are blank and are updated by hand as coefficients become available. The coefficient tables reside in the <ISCCP_ROOT>/calibration directory.

For a new satellite, a set of initial NORM and ABS table files may be created by running the script prd/src/sat_setup/CreateNewCalCoef.pl.

7.3.1 NORM file format

There is a NORM file for each SPC, year and month. The files are named NORMCOEF.spc.yymm and are text files where each line is an 80-character record. There may be more than one satellite in a file when multiple satellites are processed by the same SPC. There is typically only one set of channel records for the entire month for each satellite, as the coefficients apply to the whole month.

Example: NORMCOEF.CSU.0707 (file format is the same for both geosat and polar orbiter)

YY	MM	D1	DL	HHMMSS	HHMMSS	SATIDxxx	N	SLOPEx	INTRCPT	RMSERR	MINRNGE	MAXRNGE
07	07	01	31	000000	999999	GOES-11	1	1.140	-0.006	-1.000	0.000	0.000
07	07	01	31	000000	999999	GOES-11	2	1.037	-10.800	0.660	205.500	297.000
07	07	01	31	000000	999999	GOES-11	3	1.000	0.000	-1.000	0.000	0.000
07	07	01	31	000000	999999	GOES-11	4	1.000	0.000	-1.000	0.000	0.000
07	07	01	31	000000	999999	GOES-11	5	1.000	0.000	-1.000	0.000	0.000

YY MM = year and month

D1 D1 = first day and last day

HHMMSS HHMMSS = first and last time

N = channel number

For the first half of the project, the normalization was done once every three months and interpolated to create the NORM files for all the months in between. After that, and currently, the normalization is done every month so there is no longer need for interpolation. For H-data, the NORM files are stored in calibration/H_coefficients/norm.

7.3.2 ABS file format

There is an ABS file for each SPC, year and month. The files are named ABSCOEFS.spc.yymm and are text files where each line is an 80-character record. The polar orbiter ABS coefficients are typically for the whole month. Prior to 2010, the geosat ABS coefficients may vary from image to image so the HHMMSS columns indicates the time range. After that there is no such variation. After 2010 the geosat ABS format is similar to the polar orbiter format where there is only one entry for the month with the date range 000000-999999 instead of individual image entries.

Polar orbiter example: ABSCOEFS.NOA.0707

YY	MM	D1	DL	HHMMSS	HHMMSS	SATIDxxx	N	SLOPE _x	INTRCPT	RMSERR	MINRNGE	MAXRNGE
07	07	01	31	000000	999999	NOAA-18	1	1.075	0.000	-1.000	0.000	0.000
07	07	01	31	000000	999999	NOAA-18	2	0.977	5.667	-1.000	0.000	0.000
07	07	01	31	000000	999999	NOAA-18	3	1.000	0.000	-1.000	0.000	0.000
07	07	01	31	000000	999999	NOAA-18	4	1.000	0.000	-1.000	0.000	0.000
07	07	01	31	000000	999999	NOAA-18	5	1.000	0.000	-1.000	0.000	0.000

Geostationary example: ABSCOEFS.CSU.0707 (only first day shown as illustration)

YY	MM	D1	DL	HHMMSS	HHMMSS	SATIDxxx	N	SLOPE _x	INTRCPT	RMSERR	MINRNGE	MAXRNGE
07	07	1	1	00000	30000	GOES-11	1	1.059	0.001	-1.000	0.000	0.000
07	07	1	1	00000	30000	GOES-11	2	1.036	-10.302	-1.000	0.000	0.000
07	07	1	1	00000	30000	GOES-11	3	1.000	0.000	-1.000	0.000	0.000
07	07	1	1	00000	30000	GOES-11	4	1.000	0.000	-1.000	0.000	0.000
07	07	1	1	00000	30000	GOES-11	5	1.000	0.000	-1.000	0.000	0.000
07	07	1	1	60000	180000	GOES-11	1	1.059	0.001	-1.000	0.000	0.000
07	07	1	1	60000	180000	GOES-11	2	1.036	-9.814	-1.000	0.000	0.000
07	07	1	1	60000	180000	GOES-11	3	1.000	0.000	-1.000	0.000	0.000
07	07	1	1	60000	180000	GOES-11	4	1.000	0.000	-1.000	0.000	0.000
07	07	1	1	60000	180000	GOES-11	5	1.000	0.000	-1.000	0.000	0.000
07	07	1	1	210000	210000	GOES-11	1	1.059	0.001	-1.000	0.000	0.000
07	07	1	1	210000	210000	GOES-11	2	1.036	-10.302	-1.000	0.000	0.000
07	07	1	1	210000	210000	GOES-11	3	1.000	0.000	-1.000	0.000	0.000
07	07	1	1	210000	210000	GOES-11	4	1.000	0.000	-1.000	0.000	0.000
07	07	1	1	210000	210000	GOES-11	5	1.000	0.000	-1.000	0.000	0.000

For H-data, the ABS files are stored in calibration/H_coefficients/abs.

7.4 Initial V0 HBT files

For new satellites only the NOMINAL calibration is available. The V0 HBT files for new satellites are created using the NOMINAL calibration obtained from the input data headers and/or the SPC. Later, after a 1-2 year time record has been produced, the results of the calibration monitor program are analyzed and the NORM and ABS coefficients can then be derived and the HBT updated to a new version (see Section 8).

7.4.1 Geostationary (b1uhbt)

The initial HBT tables (V0) for geostationary satellites are created by the B1UHBT program b1uhbt.f90 which is located in prd/src. This program requires B1U data files from which it extracts the NOMINAL calibration, creates sets of calibration tables and writes them out HBT V0 files. The B1UHBT program is satellite dependent as the information in the B1U headers may vary. The code provided works for FY-2C satellite as an example. For use with other satellites it must be modified to extract the calibration information from the B1U headers as needed.

The b1uhbt step is run via initial_hbt.run script (in prd/bin) for geostationary satpos. Currently it has been tested only for satpos CMA (FY-2C). Any other satpos may require modifications as the content of the B1U header may vary.

To run:

```
$ cd wrkdir
$ ./prdev
$ cd satpos
$ initial_hbt.run . yyyy mm ddbeg ddend satpos
```

Output is written to the wrkdir, and when verified should be moved to the appropriate sub-directory of the prd/data/bt directory.

7.4.2 Polar orbiter (gachbt)

The initial HBT tables (V0) for polar orbiters are created by the GACHBT program gachbt.f90 which is located in prd/src. This program requires GAC data files from which it extracts the NOMINAL calibration, creates sets of tables and writes them out to HBT V0 files. Each HBT file contains tables for one or more orbits belonging to that 3-hour time slot.

The gachbt step is run via initial_hbt.run script (in prd/bin) for NOA and NOM satpos. In addition to the GAC data files, the SATINFO files are read to obtain bandwidth and other satellite-dependent information. See Section 7.2.3 for description of SATINFO contents.

To run, first edit the script initial_hbt.run to set the variable “isccpdir” to the parent location of the “calibration” directory, and then run as for geostationary above. Output is written to the wrkdir, and when verified should be moved to the appropriate sub-directory of the prd/data/bt directory.

8 Monitoring Calibration

This section outlines the process by which the NORM and ABS coefficients are derived. These processes are under development, and codes are located in add-on packages to be delivered separately as they may change as processes are developed.

8.1 Polar orbiter (MONITOR)

This process requires 1-2 years of GAC input data, and is under development. The results of this process are NORM and ABS coefficients for the polar orbiter VIS and IR channels.

8.2 Geostationary (NORM)

This process requires one month of B1U input data plus the contemporaneous month of afternoon orbiter GAC data. This process is under development. The results of this process are NORM coefficients for the geostationary satellite VIS and IR channels. The ABS coefficients are derived by combining the NORM and ABS coefficients from the reference polar orbiter.

8.3 Updating HBT with new coefficients

The results of the above MONITOR and NORM processes are the normalized and absolute coefficients. These coefficients are entered manually into the NORM and ABS files (created blank as part of initial satellite setup), and a new version of HBT is created for use in production (program under development). During production, the ABS coefs may be updated further, based on the overlap statistics generated during the merge step. That RECAL process is described in the following section.

9 RECAL Procedure

This section contains the basic instructions for performing re-calibration of satellites during production by producing a new version of HBT calibration tables when required by the CALTEST results. HBT Version 0 is provided through 2009 and can be updated using this process. After 2009 new HBT must be created using a separate off-line process (see Section 7.4).

9.1 Determining whether RECAL is necessary

The process of determining whether calibration corrections are necessary has three parts: d1prod, d2calcor and caltest. When d1prod runs, difference histograms are saved for areas where satellites overlap. After d1prod is finished, the d2calcor step computes the average differences for TC, TAU, RS and TS from these histograms. Then the caltest script computes offsets for the calibration tables by averaging the differences for TC and TS and for TAU (decoded as a scaled radiance) and RS. A threshold is applied such that corrections are called for only when the temperature difference is larger than 1.0K or the scaled radiance difference is larger than 0.02. Smaller differences are set to 0.0. The corrections are reported in the caltest output log. The prdrun script will scan this log and if recal is required, production will stop. At this point the recal process is done manually, a new version of HBT is created, and then production is re-started.

The output from the caltest script is contained in the log directory in the files caltest.pl.out.nn where nn is incremented each time caltest is run. For each satellite position this file will indicate either “OK” or “RECAL REQUIRED”. In addition, there may be a “WARNING” about opposite signs on the corrections which should be investigated.

9.2 Computing new IR and VIS offsets

The columns labeled “Absolute Calib Adjustment” contain the IR and VIS offset numbers for the current run. Any offsets that are non-zero (i.e. above the threshold) will trigger the “RECAL REQUIRED” condition. The RECAL process must iterate until no further corrections are needed. Each time it iterates, the offsets must be added to the previously reported offsets to obtain a cumulative offset to apply to the original HBT tables. This means that all the caltest.pl.out log files must be looked at each time and the offsets added up to single number.

9.3 Investigating warnings about opposite signs

If the log indicates “WARNING: IR opposite signs” that means that caltest.pl has found that the TC and TS corrections are of opposite sign, whether or not the offsets are non-zero. Similarly, a “WARNING: VIS opposite signs” means that TAU and RS corrections are of opposite sign. This condition should be investigated. The first thing to check would be to look at the input calibration corrections to make sure they were correct. Other investigation techniques include looking at the spatial patterns of the differences between the satellites and making regression plots. Depending on results of the investigation, calibration adjustments may or may not be required. This will be further discussed in troubleshooting training.

9.4 Generating a new version of HBT tables

Go to the satpos sub-directory in the wrkdir for the run.

From this satpos sub-directory run the btcacror process to apply the offsets and generate new version of HBT tables:

```
$ btcacror.run . yyyy mm satpos input_version ir_offset vis_offset |tee logfile
```

Where:

“.” indicates it will run in the current directory

“yyyy mm” is the year and month

“satpos” is the 3-letter satellite position

“input_version” is 0 for starting with initial HBT files (it is possible to start with higher versions and apply offsets to those instead)

“ir_offset vis_offset” calculated by you from cumulative caltest results

“logfile” is output file specified by you

The script will ask you to verify the input version and offset arguments, so use “|tee” rather than “>” to redirect the output, so that you can see the script prompt you for (y/n) input. If the new HBT directory already exists (for example if you forgot to move a previous run to the production directory) the script will ask you whether you want to overwrite or not.

When this runs, it will create a directory and write the new HBT files into it, with version number incremented. HBT files are only created for existing images. The program tries to find existing HBT for each 3-hour timeslot, if it is not available a warning is printed and the program continues. Make sure the number of missing files is what is expected.

Also written to the new version directory is a VERSION_INFO.txt file containing a record of what was done to produce this version (input version number, offsets applied) and the date/time it was run.

Spot check the new HBT files to make sure they are correct. If okay, move the whole HBT directory that was created to the production data directory prd/data/bt/satpos/sss-yyy. It is okay to have both V0 and V1 there at the same time because production will always use the file with the largest version number.

9.5 Re-running the satellites

Once the newly produced HBT version directory has been installed in the prd/data directory tree, then re-schedule full production for the satpos that are being corrected (b4 through ds), and also the merge products (d1, d2 etc.) and start prdrun. Production will now use the new HBT V1 files (largest version number found), and the caltest script will check the results again.

9.6 Finishing production

If caltest indicates that further corrections are required, repeat the process using a cumulative offset by adding together all the offsets from the caltest log files so far. Keep repeating the process until all the caltest adjustments are 0.0. You do not need to keep intermediate HBT that

is created, only the final one. If caltest finds no further adjustments then production will continue to the end products.

9.7 Updating the ABS coefficient files

In the final HBT directories for each satpos, the VERSION_INFO.txt file will contain the final offset values that were used. In the case of the final run, that is the accumulated offset number (otherwise it wouldn't be the final run), and no further modification is required. The final directory, including the VERSION_INFO file, should be archived and the offsets applied to the ABS coefficient files to produce a new version.

To do this automatically, an example script is provided (prd/bin/calcoefs_update.pl) that will read the VERSION_INFO file, apply the offsets to the current ABS file, and update the ABS file in an svn version control system. This script can be modified as needed to work with your specific system setup.

9.8 Sample RECAL scenario

Suppose you have run production for July 2007 using HBT version 0. After the merge step production has stopped indicating "RECAL REQUIRED". Suppose the logfile for caltest (log/caltest.pl.out.0) looks like this (these are not the real numbers, just an example):

```

/* Calibration Corrections          /* Absolute Calib Adjustment
/*   TC      TS      TAU      RS   /*          IR      VIS
/* -----
/*   1.565   1.576   0.005   0.016 /* 1 GMS /* 1.00   0.00 RECAL REQUIRED
/*  -1.328  -0.084   0.045   0.075 /* 2 MET /* -1.50   0.05 RECAL REQUIRED
/*   0.761   0.046   0.008   0.004 /* 3 GOW /* 0.00   0.00 OK
/*   0.003   0.424  -0.024  -0.005 /* 4 GOE /* 0.00   0.00 OK
/*  -2.724  -1.941   0.010   0.006 /* 5 INS /* -1.50   0.00 RECAL REQUIRED
/*   9.000   9.000   9.000   9.000 /* 6 NOA /* 0.00   0.00 OK
/*   0.007   0.168   0.005   0.028 /* 7 NOM /* 0.00   0.00 OK

```

This shows you that the RECAL process is required for the GMS, MET and INS satellite positions. The NOA satpos is the standard so it is always filled with 9.000 and adjustments are always 0.0.

So you would run btcalcor for those three satpos like this:

Set environment

```

$ cd isccp_root/prd/wrkdirs/2007_07
$ . ./prdens

```

Recal GMS satpos

```

$ cd isccp_root/prd/wrkdirs/2007_07/gms
$ btcalcor.run . 2007 7 gms 0 1.0 0.0 |tee recal_gms.txt
$ mv ISCCP.HBT.1.MTS1.2007.07.JMA isccp_root/prd/data/bt/gms/MTS1.2007

```

Recal MET satpos

```
$ cd isccp_root/prd/wrkdirs/2007_07/met
$ btcalcor.run . 2007 7 met 0 -1.5 0.05 |tee recal_met.txt
$ mv ISCCP.HBT.1.MET9.2007.07.EUM isccp_root/prd/data/bt/met/MET9.2007
```

Recal INS satpos

```
$ cd isccp_root/prd/wrkdirs/2007_07/ins
$ btcalcor.run . 2007 7 ins 0 -1.5 0.00 |tee recal_ins.txt
$ mv ISCCP.HBT.1.MET7.2007.07.MTI isccp_root/prd/data/bt/ins/MET7.2007
```

Use “|tee” rather than redirect “>” so that you can see the script prompt you for confirmation and enter “y” or “n” when requested.

Then you would re-run production for these three satpos. Using prdsched, schedule full production for GMS, MET and INS, and then check all the boxes for the merged products (d1, d2 etc) and start prdrun.

Suppose it stops again saying "RECAL REQUIRED", you would look at the new logfile file for caltest (log/caltest.pl.out.1) and suppose it looks like this:

/*	TC	TS	TAU	RS	/*	IR	VIS	
/*	0.065	0.076	0.005	0.016	/* 1 GMS */	0.00	0.00	OK
/*	-0.328	-0.084	-0.015	-0.012	/* 2 MET */	0.00	-0.01	RECAL REQUIRED
	0.761	0.046	0.008	0.004	/* 3 GOW */	0.00	0.00	OK
	0.003	0.424	-0.024	-0.005	/* 4 GOE */	0.00	0.00	OK
	-0.224	-0.041	0.010	0.006	/* 5 INS */	0.00	0.00	OK
	9.000	9.000	9.000	9.000	/* 6 NOA */	0.00	0.00	OK
	0.007	0.168	0.005	0.028	/* 7 NOM */	0.00	0.00	OK

This indicates that a further correction is necessary for the MET satpos. So to apply to the original V.0 add the offsets together, and redo the HBT again. For IR, the first caltest showed an adjustment of -1.5 and the second showed 0.0 so the total adjustment is -1.5. For VIS, the first caltest showed an adjustment of 0.05 and the second showed -0.01 so the total adjustment now is 0.04. These offsets will be applied to the original V0 to create a new V1 that will replace the one created in the first iteration.

Set environment

```
$ cd isccp_root/prd/wrkdirs/2007_07
$ . ./prdens
```

Recal MET satpos

```
$ cd isccp_root/prd/wrkdirs/2007_07/met
$ btcalcor.run . 2007 7 met 0 -1.5 0.04 |tee recal_met.txt
```

Replace the V1 table directory in the production directory

```
$ rm -r isccp_root/prd/data/bt/met/MET9.2007/ISCCP.HBT.1.MET9.2007.07.EUM
$ mv ISCCP.HBT.1.MET9.2007.07.EUM isccp_root/prd/data/bt/met/MET9.2007
```

Now schedule full production for MET satpos only this time, and for all the merged products again, and start prdrun.

Suppose this time it does not stop, and goes on to produce the final products. The new caltest logfile (log/caltest.pl.out.2) looks like this:

```

/*      TC      TS      TAU      RS      */
/*      -----      * /
0.065   0.076   0.005   0.016 /* 1 GMS */
-0.328  -0.084  -0.005  -0.002 /* 2 MET */
0.761   0.046   0.008   0.004 /* 3 GOW */
0.003   0.424  -0.024  -0.005 /* 4 GOE */
-0.224  -0.041   0.010   0.006 /* 5 INS */
9.000   9.000   9.000   9.000 /* 6 NOA */
0.007   0.168   0.005   0.028 /* 7 NOM */
IR      VIS
-----
0.00    0.00    OK

```

All adjustments are now 0.0 so the run is finished and you have the final set of HBT V1.

9.9 HBT format

There is one file for each 3-hour time slot, 8 per day, for each satellite. For geo-stationary satellites (geosat), each file contains tables for one image. For polar orbiters (leosat), each file contains multiple sets of tables, for as many orbits as occur in the 3-hour time slot.

The HBT files provided were created from BT files, which corresponded to B3 images processed. In B3 data, the channels are re-ordered to a consistent ISCCP order, so the HBT channel order is called “B3 order” below. When processing B1U data, the B1U channels are reordered to ISCCP order as they are read in. For polar orbiter data, the orbit begin and end times in the HBT files are the actual begin and end times of the B3 orbit, which was trimmed from the GAC orbit, so will not match exactly. The GAC orbit begin/end time range will contain the B3 orbit begin/end time range for matching orbits.

Each geosat image, or leosat orbit in the file, has a series of table sets, each consisting of a one-line header followed by the 6 calibration tables. Table sets are repeated for each channel that was in the B3 data file. For files with multiple orbits, the whole series (header+tables repeated for each channel) is then repeated for each orbit in the 3-hour time slot.

Each header line contains the following text fields in the format (A8,I9,2I2,2I7,6I2):

vsatid	8-char abbreviated satellite name (just an identifier, not used in production)
idate	Date (yymmddgg) - it is ok if there is no leading zero on the year
ichan	B3 channel number
iavail	Table set availability flag (0/1) 0 = tables not available (next line will be another header) 1 = tables are available (next 256 lines are the table entries)
itime1	Beginning time (hhmmss)
itime2	Ending time (hhmmss) Geosats: itime1 and itime2 are identical, it is GMT slot (000000 – 210000) Leosats: itime1 and itime2 are actual begin and end times of the B3 orbit
tabflg(6)	Table status flag (0/1)

0 = tables are empty or bad
1 = tables are full and ok

After the header line, there are 256 lines of 6 columns each. The lines correspond to the count values 0-255 (count 255 is reserved to indicate “undefined”), and the 6 columns are the calibration tables in the format (6F8.3):

Column 1 = Nominal physical radiance

Column 2 = Normalized physical radiance

Column 3 = Absolute physical radiance

Column 4 = Nominal scaled radiance or temperature, depending on which channel it is

Column 5 = Normalized scaled radiance or temperature, depending on which channel it is

Column 6 = Absolute scaled radiance or temperature, depending on which channel it is

This header+tables set is repeated for each B3 channel of each image/orbit in the 3-hour time slot. This HBT format is the same as it was for the BT dataset, and can be read/written using functions in prd/src/btdata_module.f90.

END OF DOCUMENT