

Radiance Simulator v1.0 User Guide

Andrew Smith, Met Office, UK

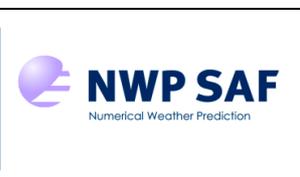
This documentation was developed within the context of the EUMETSAT Satellite Application Facility on Numerical Weather Prediction (NWP SAF), under the Cooperation Agreement dated 29 June 2011, between EUMETSAT and the Met Office, UK, by one or more partners within the NWP SAF. The partners in the NWP SAF are the Met Office, ECMWF, KNMI and Météo France.

COPYRIGHT 2014, EUMETSAT, ALL RIGHTS RESERVED.

Change record			
Version	Date	Author / changed by	Remarks
0.1	21/08/2014	A. Smith	Final draft
0.2	10/09/2014	A. Smith	Updated after beta-testing
0.3	17/10/2014	A. Smith	Updated for DRI
1.0	17/11/2014	A. Smith	Version updated for release

Table of Contents

1.	INTRODUCTION	3
2.	INSTALLATION	3
2.1	System Requirements.....	3
2.2	External Libraries.....	3
2.3	Compiling the code.....	4
3.	RUNNING THE CODE	5
4.	INPUT/OUTPUT FILES.....	6
4.1	Configuration namelist file	6
4.2	Observation data file	7
4.3	Model data files.....	8
4.3.1	UM Fieldsfiles / PP files (<i>file type 0</i>).....	9
4.3.2	GRIB files (<i>file type 1</i>)	10
4.3.3	NWP-SAF 60L and 91L profile datasets (<i>file types 2,3</i>)	11
4.4	Output file.....	12
4.4.1	Naming convention.....	12
4.4.2	Contents	12
5.	PROCESSING DETAILS.....	14
5.1	Use of observation information.....	14
5.1.1	Horizontal interpolation.....	14
5.2	Non-standard grids.....	15
5.2.1	Coordinate rotation.....	15
5.3	Field conversions.....	16
5.3.1	Potential temperature conversion	16
5.3.2	Humidity conversion	16
5.4	Pressure levels	17
5.5	Surface type.....	18
5.6	Emissivity values	18
5.7	Running RTTOV.....	18
5.8	Quality control	18
6.	LIMITATIONS AND KNOWN ISSUES.....	19

		Radiance Simulator v1.0 User Guide	Doc ID : NWPSAF-MO-DS-033 Version : 1.0 Date : 17/11/2014
--	--	---	---

1. INTRODUCTION

This document describes the NWP SAF Radiance Simulator code which is designed to output simulated satellite instrument data from NWP model fields. Details of how to install and run the code, input/output requirements and a description of the processing that is performed are included.

The code is written primarily in Fortran-90. Users will need some knowledge of Fortran namelists and netCDF files and should ideally understand the capabilities of the RTTOV radiative transfer model in order to use the Radiance Simulator but in-depth knowledge is not required.

2. INSTALLATION

2.1 System Requirements

- Unix or Linux operating system
- Fortran-90 compiler with some basic F2003 support
- 10 MB free disk space for installation

The code has been tested successfully in one or more standard configurations after installation with each of the following compilers:

- ifort - v12.0.4
- pgfortran - v11.7-0
- gfortran - v4.4.6
- xlf – v12.1.0.9

Newer versions of these compilers are expected to be compatible. It may also work with some older versions but these have not been tested and are not supported. The NAG compiler *nagfor* is not supported as byte-swapping I/O is not available in the installed version (5.2) on the development platform, however this is available in version 5.3 and users may attempt to install with *nagfor* if they have access to this version (a compiler configuration file is provided).

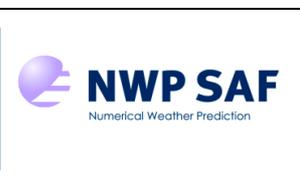
2.2 External Libraries

Fortran versions of the following libraries are required. Each can be downloaded from the given locations if not already available.

- RTTOV (version 11.1 or later)

Available from <http://nwpsaf.eu/index.html>

Coefficient files will also be needed. These can be downloaded from the same location. Additional IR cloudy and MW scattering (Mie) coefficient files will be needed if using either

		Radiance Simulator v1.0 User Guide	Doc ID : NWPSAF-MO-DS-033 Version : 1.0 Date : 17/11/2014
--	--	---	---

one of those options. The standard coefficients are for clear-sky simulations only. The unformatted .dat files are required in each case.

- GRIB API (version 1.10.0 or later)

Available from <https://software.ecmwf.int/wiki/display/GRIB/Home>

- netCDF (version 4.0 or later)

Available from <http://www.unidata.ucar.edu/software/netcdf/>

The location of the following libraries may be needed in some cases but this is system dependent. They are used by one or more of the above libraries so there should be no need to download new versions of these. If installation fails because one of these libraries is missing then you will need to add it to the user configuration (see below).

- HDF5 (version 1.8.7 or later, used by netCDF and RTTOV)

Available from <http://www.hdfgroup.org/HDF5/>

- Jasper (used by GRIB API)

2.3 Compiling the code

All of the Radiance Simulator code is contained in the distribution file `radsim.tar.gz`. Access to some external libraries (see above) is also required for compilation.

Installation consists of the following steps.

1. Unpack the distribution file in the installation directory

```
tar -xzf radsim.tar.gz
```

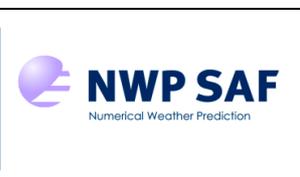
Or...

```
gunzip radsim.tar.gz; tar -xf radsim.tar
```

...if the `-z` flag is not supported.

The directory should now contain, in addition, the following files

```
build/*
etc/*
src/*
radsim_install
readme.txt
user.cfg
```

		Radiance Simulator v1.0 User Guide	Doc ID : NWPSAF-MO-DS-033 Version : 1.0 Date : 17/11/2014
--	--	---	---

2. Edit the user.cfg file to supply paths to external libraries. Also set the name of the compiler you wish to use (one of ifort, gfortran, pgfortran or xlf). The following environment variables should then be defined:

```
FC                # Name of Fortran compiler
GRIB_API_DIR     # Path to GRIB_API installation directory
NETCDF_DIR       # Path to netCDF installation directory
RTTOV_DIR        # Path to RTTOV installation directory
```

You may also need to set (this is system dependent)

```
HDF_DIR          # Path to HDF5 installation directory
JASPER_DIR       # Path to Jasper installation directory
```

Each library path should contain the standard lib and include subdirectories and in the case of RTTOV a mod subdirectory.

3. Run the install script.

```
radsim_install
```

The directory should now contain, in addition, the following files

```
bin/radsim.exe
bin/radsim_run # Wrapper script for radsim.exe
```

Note on compiler options:

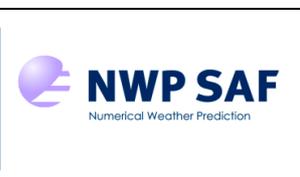
All currently supported input profile datasets are written in big-endian byte order (with the exception of GRIB which is a portable format). It is assumed in the case of the first 3 compilers listed in the System Requirements section above that these will be used on a little-endian platform such as a Linux PC and therefore byte-swapping for I/O has been included as a compiler flag in each case. Users may change compiler options by editing the relevant compiler configuration file in the `build` subdirectory. For the *ifort* compiler, the `F_UFMTENDIAN` environment variable may be used instead to override the I/O byte-order assumption.

3. RUNNING THE CODE

The Radiance Simulator can be run by issuing the following command (in this case from the radsim installation directory)

```
bin/radsim_run radsim_cfg.nl
```

`radsim_cfg.nl` is a Fortran namelist file containing configuration options, including the location of input and output files and the instrument to be simulated. Users are recommended to take a copy of the example file `etc/radsim_cfg_example.nl` and

		Radiance Simulator v1.0 User Guide	Doc ID : NWPSAF-MO-DS-033 Version : 1.0 Date : 17/11/2014
--	--	---	---

edit according to their requirements. Full details of the available options are given in the section below. At the very least, the following need to be set

- Path to profile dataset
- Path to directory containing RTTOV coefficient files
- Satellite instrument identifiers

4. INPUT/OUTPUT FILES

4.1 Configuration namelist file

This file contains a single Fortran namelist `radsim_nl`. The namelist consists of the following variables (Fortran intrinsic type and inclusion status or default value are given in parentheses, compulsory variables also highlighted in bold):

Input data variables:

- `obs_datafile` (*character, optional*) - Path to observation data file. This allows the simulation to be performed at observation locations. The file format is described in section 4.2.
- **`model_datafile`** (*character, compulsory*) - Path to model data file. Several formats are supported (see section 4.3)
- **`model_filetype`** (*integer, compulsory*) – Model file type identifier. See section 4.3.
- **`rttov_coefs_dir`** (*character, compulsory*) – Path to RTTOV (clear-sky) coefficients directory.
- `rttov_mietable_dir` (*character, optional*) – Path to RTTOV Mie table coefficients directory for use with the `run_scatt` option.
- `rttov_scald_dir` (*character, optional*) – Path to RTTOV IR scattering coefficients directory for use with the `ir_addclouds` option.

Satellite instrument variables:

- **`platform`** (*character, compulsory*) – RTTOV platform identifier. This is the same as used in the RTTOV coefficient file name.
- **`inst`** (*character, compulsory*) – RTTOV instrument identifier. This is the same as used in the RTTOV coefficient file name.
- **`satid`** (*integer, compulsory*) – RTTOV satellite identifier. This is the same as used in the RTTOV coefficient file name.
- `channels` (*integer array, optional*) – List of channel numbers to use. If omitted or if all values are 0 then all channels will be used.

General control variables:

- `run_scatt` (*logical, default=F*) – Run MW scattering code RTTOV-Scatt. This requires extra fields in the input model data file (see section 4.3) and RTTOV Mie tables.
- `output_mode` (*integer, default=1*) – Diagnostic output mode. Three levels are defined: 1=Normal, 2=Verbose, 3=Debug.

		Radiance Simulator v1.0 User Guide	Doc ID : NWPSAF-MO-DS-033 Version : 1.0 Date : 17/11/2014
---	---	---	---

- `seaice_thresh` (*real, default=0.5*) – Separation threshold for sea or seaice surface type classification

RTTOV-specific options (these are commuted directly to RTTOV equivalents):

- `ozone_data` (*logical, default=F*) – Use input ozone profile
- `clw_data` (*logical, default=T*) – Use input cloud-liquid-water profile (applies to MW simulation only)
- `use_totalice` (*logical, default=F*) – Use a total ice field or else separate into cloud-ice-water and precipitating ice (applies only if `run_scatt=T`). Note that this may depend on the source of the model data (see also section 4.3).
- `ir_addclouds` (*logical, default=F*) – Run IR cloud scattering code. Note that this option also requires additional RTTOV coefficient files to be specified in the `rttov_scclld_dir` variable.
- `ircloud_idg` (*integer, default=4*) – Size distribution to use with `ir_addclouds=T`. See RTTOV v11 Users Guide for details.
- `ircloud_ish` (*integer, default=3*) – Ice crystal shape for use with `ir_addclouds=T`. See RTTOV v11 Users Guide for details.
- `fastem_version` – Version of the *Fastem* microwave emissivity model to use

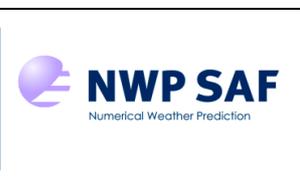
Output control variables:

- `output_dir` (*character, optional*) – Path to output directory. The default is to use the current directory. It should be noted that output files can be very large. Further details on output are given in section 4.4.
- `write_radiances` (*logical, default=F*) – Write radiances to the output file. The default is to write brightness temperatures.
- `write_emiss` (*logical, default=F*) – Write emissivity values to the output file.
- `write_profiles` (*logical, default=F*) – Write model profile data to the output file, including all fields that were used in the simulation (fields will be interpolated to obs positions if that option has been used).
- `write_trans` (*logical, default=F*) – Write layer transmittances to the output file. This significantly increases the size of the output file and the run time.
- `write_tjac` (*logical, default=F*) - Write temperature Jacobians to the output file. This significantly increases the size of the output file and the run time.
- `write_qjac` (*logical, default=F*) - Write humidity Jacobians to the output file. This significantly increases the size of the output file and the run time.
- `write_o3jac` (*logical, default=F*) - Write ozone Jacobians to the output file. This significantly increases the size of the output file and the run time.

4.2 Observation data file

This file contains observation metadata. The purpose is to allow simulations to be performed at specific locations with realistic satellite viewing geometry (see section 5.1 for more details).

The file is a Fortran formatted file containing the data for one observation point per line. For each observation the following need to be provided (all are *real* values):

		Radiance Simulator v1.0 User Guide	Doc ID : NWPSAF-MO-DS-033 Version : 1.0 Date : 17/11/2014
--	--	---	---

Latitude (degrees, range -90.0 -> 90.0)

Longitude (degrees, range -180.0 -> 180.0, or 0.0 -> 360.0)

Surface elevation (m)

Land-sea mask (0.0 -> 1.0) – this is effectively a land fraction so 0.0=sea, 1.0=land

Satellite zenith angle (degrees)

Observation data are preceded by a short header section. Users should view the `obsdata_example.txt` file in the `etc` subdirectory of the installation directory to see how the file is laid out.

4.3 Model data files

Atmospheric profile data can be obtained from a number of sources. The following are currently supported (the file type code used in the configuration namelist is given in parentheses):

- Met Office UM fieldsfiles/PP files (*file type 0*)
- GRIB files (*file type 1*)
- NWP SAF 60L profile dataset from ECMWF analyses (*file type 2*)
- NWP SAF 91L profile dataset from ECMWF short-range forecasts (*file type 3*)

Regardless of the data source, the following fields must be available in the file, either directly or in an equivalent form that can be converted easily. Available conversions are described in section 5.3. Further details on file contents can be found in the following subsections.

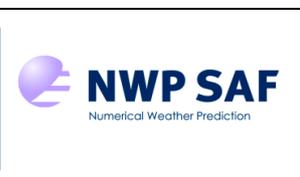
Required fields on levels (the same levels must be used in each case)

- Pressure
- Temperature
- Specific humidity

Required surface fields

- Pressure
- Temperature (at 2m)
- Specific humidity (at 2m)
- Skin temperature
- Surface wind speed (as separate latitudinal and longitudinal or *u* and *v* components)
- Land-sea mask
- Orographic height
- Sea-ice fraction (see note below)

Note: sea-ice fraction is not compulsory (the simulations will still run) because it may not be a relevant parameter for certain profile data sets e.g., for a limited area model that does not cover high latitudes, but it should always be provided if possible.

		Radiance Simulator v1.0 User Guide	Doc ID : NWPSAF-MO-DS-033 Version : 1.0 Date : 17/11/2014
--	--	---	---

Required field for option `clw_data=T`

- Specific liquid water content (cloud-liquid-water)

Required fields for option `run_scatt=T`

- Specific liquid water content (cloud-liquid-water)
- Specific ice content (cloud-ice-water)
- Rain rate
- Snow rate

Required fields for option `ir_addclouds=T`

- Specific liquid water content (cloud-liquid-water)
- Specific ice content (cloud-ice-water)
- Area cloud fraction
- Liquid cloud fraction
- Ice cloud fraction

Possible supplementary fields for option `ir_addclouds=T`

- Convective cloud amount
- Convective cloud water content

The use of these supplementary fields depends on the nature of the clouds defined in the NWP model that is the source of the dataset. In some cases, cloud is not separated into convective and large-scale types and therefore convective cloud fields may not be available.

4.3.1 UM Fieldsfiles / PP files (*file type 0*)

These must contain *unpacked* data. The Radiance Simulator will detect if this is not the case and exit without further processing. If access to UM utility scripts is available, then the *iee* tool can be used to unpack the data prior to running the simulation, otherwise you will need to request an unpacked file from your data provider (warning note: unpacked files may be up to an order of magnitude larger than packed files).

In the case of fieldsfiles, only 64-bit files are currently supported. PP files must have the '.pp' suffix in the file name. Both 32-bit and 64-bit PP files can be used although the use of 64-bit files is deprecated on Met Office systems.

The file should contain the STASH items listed in the following table. They should be valid at the same data time. If STASH items are present for more than one data time then the earliest data time will be selected.

STASH code	Description
4 (or 16004)	Theta (or temperature)
10	Specific humidity
24	Surface skin temperature
31	Sea-ice fraction
33	Orography
407 (or 408)	Pressure levels
409	Surface pressure
3209	10m wind (u-component)
3210	10m wind (v-component)
3236	Temperature at 1.5m
3245	Relative humidity at 1.5m

The following STASH items may also be required depending on what options are selected.

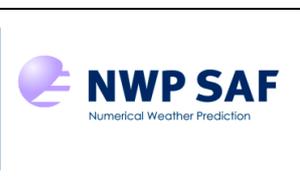
STASH code	Description	Usage notes
12	Specific ice content (qcf)	Required for <i>run_scatt</i> (see note 1) and <i>ir_addclouds</i>
186	Rain rate	Optional for <i>run_scatt</i> (see note 2)
187	Snow rate	Optional for <i>run_scatt</i> (see note 1)
254	Specific liquid water content (qcl)	Required for <i>clw_data</i> , <i>run_scatt</i> , <i>ir_addclouds</i>
265	Area cloud fraction	Required for <i>ir_addclouds</i>
267	Liquid cloud fraction	Required for <i>ir_addclouds</i>
268	Ice cloud fraction	Required for <i>ir_addclouds</i>
5212	Convective cloud amount	Optional for <i>ir_addclouds</i> (see note 3)
5213	Convective cloud condensed water	

Notes:

1. UM diagnostic output usually includes all ice, including precipitating ice, in the *qcf* field (STASH code 12), therefore it is recommended that the *use_totalice* option is switched on if running with the *run_scatt* option. The capability to use snow rate has been included but is not currently applicable.
2. Rain is not included in the liquid water field, it has to be added separately. Although desirable, it is not considered compulsory - the simulation can still be run without a rain-rate field if it's not available.
3. Convective cloud as a separate diagnostic is not available from models using the PC2 cloud scheme (this includes the global model) as the liquid water field *qcl* contains all cloud.

4.3.2 GRIB files (*file type 1*)

Only GRIB files originating from ECMWF are supported as this was the only data source available for testing the current release. Support for other sources may be added in later

		Radiance Simulator v1.0 User Guide	Doc ID : NWPSAF-MO-DS-033 Version : 1.0 Date : 17/11/2014
--	--	---	---

releases but this is likely to require additional code to extract the necessary pressure-level information.

The file should contain the following fields as defined by the GRIB-API *ParamId* key value.

ParamId	Description
3 (or 130)	Theta (or temperature)
133 (or 157)	Specific (or relative) humidity
235	Surface skin temperature
31	Sea-ice fraction
129	Geopotential (for orography)
54	Pressure levels (optional, see note 1 below)
134 (or 152)	Surface pressure
165	10m wind (u-component)
166	10m wind (v-component)
167	Temperature at 2m
168	Dewpoint temperature at 2m

The following fields may also be required depending on which options are selected.

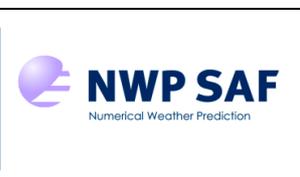
ParamId	Description	Usage notes
247	Specific ice content (qcf)	Required for <i>run_scatt</i> (see note 2) and <i>ir_addclouds</i> (see note 4)
75	Rain (mixing ratio)	Optional for <i>run_scatt</i> (see note 3)
76	Snow (mixing ratio)	Optional for <i>run_scatt</i> (see note 2)
246	Specific liquid water content (qcl)	Required for <i>clw_data</i> , <i>run_scatt</i> , <i>ir_addclouds</i> (see note 4)
248	Area cloud fraction	Required for <i>ir_addclouds</i>

Notes:

1. Pressure levels are usually not included explicitly and need to be reconstructed from the surface pressure field and a set of coefficients.
2. Users will have to decide whether the *use_totalice* option is applicable for the data they are using, i.e., whether the ice field contains all ice or whether a snow field should be included separately.
3. The simulations may be run with or without the rain field.
4. It is assumed that liquid and ice water contents are representative of the cloud and not layer mean values as is the case with UM fieldsfiles (mean values require, in addition, the ice and liquid volume cloud fraction variables in order to obtain mass per unit volume).

4.3.3 NWP-SAF 60L and 91L profile datasets (*file types 2,3*)

Both of these datasets contain sufficient information to run the optional IR or MW cloudy (scattering) simulations if desired. In the case of MW scattering, the *use_totalice* option is not available as these datasets contain separate cloud-ice and precipitating-ice fields.

		Radiance Simulator v1.0 User Guide	Doc ID : NWPSAF-MO-DS-033 Version : 1.0 Date : 17/11/2014
--	--	---	---

Also, because they are neither arranged on a regular lat-lon grid or representative of a single data-time, they cannot be interpolated to observation positions. Therefore, simulations are performed at profile locations. If an observation data file is specified in the configuration file then it is simply ignored in this case.

Users should consult the accompanying documentation for further details about the content of these datasets.

4.4 Output file

4.4.1 Naming convention

Output data is written to a netCDF file with the following naming convention:

```
radsim-<platform>_<satid>_<instrument>-<datetime>.nc
```

e.g.,

```
radsim-metop_2_amsua-2014010100.nc
```

The data-time suffix (always of the form YYYYMMDDHH) is optional and will be omitted, along with the last '-' separator, if the profile dataset is not related to a single validity time (e.g., the NWP SAF datasets).

4.4.2 Contents

The following *attributes* are written to the file

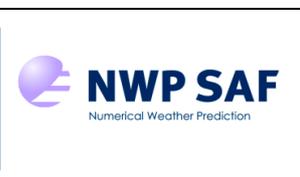
- *platform, satid, instrument* – character strings identifying the simulated instrument.
- *channels* – integer array of the channels used.
- *validity_time* – 5 element integer array containing year, month, day, hour, minute. This will not be written if the profile data have variable validity times.
- *wavenumbers* – array of wavenumber values corresponding to the channels used, taken from the RTTOV coefficient file.

Most output fields are floating point arrays with the first dimension the number of simulated observations (labelled *obs* in the file). Other dimensions may be *levels* and *channels*.

For the purpose of the simulation, fields are divided into observation and model fields. The former consist of any information in the observation data file (if applicable), plus any simulated data. The latter consist of model profile data. By default, **all observation fields are written to the file**. Observation fields are listed below:

Fields that are always present:

- *lat, lon* – observation coordinates
- *satzen, satazim* – satellite zenith, azimuth angles
- *ism* – land-sea mask (real value 0.0=sea, 1.0=land)

		Radiance Simulator v1.0 User Guide	Doc ID : NWPSAF-MO-DS-033 Version : 1.0 Date : 17/11/2014
--	--	---	---

- *surface* – RTTOV surface type (land, sea or seaice)
- *qcflags* – quality control flags (see section 5.8 below for more details)
- *qcinfo* – additional quality control information
- *bt* OR *radiance* – brightness temperatures or radiances; only one of these is written
- *emiss* – emissivities

Fields that may be present depending on the options chosen:

- *tjac*, *qjac*, *o3jac* – temperature, humidity and ozone Jacobians
- *trans* – layer transmittances

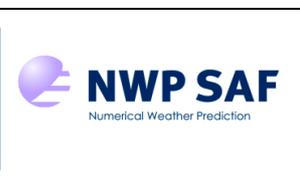
Model fields are not written out by default but they can be included by use of the *write_profiles* option in the namelist file. All fields are those used in the simulation (before conversion to RTTOV units), therefore if simulating at observation positions they are interpolated values not input values, and there will be as many data points in each field (per level) as there are observations. Also, any input fields that are converted to a required field are discarded after conversion and therefore not available (e.g., dewpoint temperature is converted to specific humidity and then discarded). The full list of possible fields is as follows:

Fields that are always present if using the *write_profiles* option:

- *lsm* – land-sea mask (0-1)
- *zsurf* – surface height (m)
- *seaice* – seaice fraction (0-1)
- *pstar* – surface pressure (Pa)
- *t2* – temperature at 2m (K)
- *q2* – specific humidity at 2m (kg/kg)
- *tskin* – surface skin temperature (K)
- *u10* – 10m wind u-component (m/s)
- *v10* – 10m wind v-component (m/s)
- *p* – pressure levels (Pa)
- *t* – temperature profiles (K)
- *q* – specific humidity profiles (kg/kg)

Fields that may be present depending on the options chosen and the contents of the data file:

- *ph* – pressure at half-levels (Pa)
- *z* – height at pressure levels (m)
- *clw* – cloud liquid water profiles (kg/kg)
- *ciw* – cloud ice water profiles (kg/kg)
- *rain* – rain profiles (either kg/m²/s or kg/kg, depending on input profile)
- *snow* – snow profiles (either kg/m²/s or kg/kg, depending on input profile)
- *cfrac* – cloud fraction profiles (0-1)
- *cfrac_liq* – liquid cloud fraction profiles (0-1)

		Radiance Simulator v1.0 User Guide	Doc ID : NWPSAF-MO-DS-033 Version : 1.0 Date : 17/11/2014
--	--	---	---

- *cfrac_ice* – ice cloud fraction profiles (0-1)
- *cfrac_conv* – convective cloud fraction profiles (0-1)
- *conv_cloud* – convective cloud amount profiles (kg/kg)
- *o3* – ozone profiles (kg/kg)

The size of the output files obviously depends on the configuration, notably the number of instrument channels, the number of profiles and whether some of the more expensive options are chosen such as the production of Jacobians or transmittances. An AMSU-A simulation in a simple configuration may only require around 5MB, whereas a similar configuration for IASI using all 8461 channels may require considerably more than 2GB.

5. PROCESSING DETAILS

In simple terms, the function of the Radiance Simulator is to read a model data file, set up and run a radiative transfer model (RTTOV) using these data as input, and collect the results at the end. A basic enhancement is the use of observation location information but otherwise, any additional processing exists solely to assist in these steps. A description of the additional processing is given in the following subsections.

5.1 Use of observation information

Observation meta-data are provided via the observation data file which is described in section 4.2. The purpose is to allow more realistic simulations of a satellite instrument in-flight by specifying observation positions and the viewing angle, rather than doing the calculations at model grid points and assuming nadir view angle. The model fields must satisfy two criteria in this case – they must be defined for a single data time and they must be arranged on a regular lat-lon grid (this includes regular grids with a rotated pole – see section 5.2).

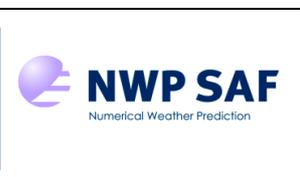
Assuming the above criteria are met, model fields are interpolated horizontally to observation positions, with the exception of any fields provided in the observation data file (this currently only includes surface type and surface height). It is assumed that the user has access to better information for these at observation locations.

5.1.1 Horizontal interpolation

A standard bi-linear interpolation technique is used. The interpolated value is a weighted combination of the values at the 4 nearest neighbouring points that form an enclosing rectangular *grid cell*.

A regular lat-lon grid is defined with a datum coordinate and values for the lat and lon intervals between adjacent points. For a point *P*, the grid cell index in each axis direction is then the difference between the datum coordinate and the position of *P*, divided by the interval.

Model fields are stored as 1D-arrays so the single combined grid cell index is dependent on the ordering of grid points in the field. A regular grid is assumed by default to be

		Radiance Simulator v1.0 User Guide	Doc ID : NWPSAF-MO-DS-033 Version : 1.0 Date : 17/11/2014
--	--	---	---

arranged row-wise, i.e., the data for the first row is followed by the data for the second row and so-on, however column-wise ordering is also allowed.

For a grid cell with vertices $[(x,y), (x+dx,y), (x,y+dy), (x+dx, y+dy)]$ and corresponding field values $[a_1, a_2, a_3, a_4]$, the interpolated field value b is given by

$$b = (1 - w_x)(1 - w_y)a_1 + w_x(1 - w_y)a_2 + (1 - w_x)w_y a_3 + w_x w_y a_4$$

Where the weights w_x and w_y are

$$w_x = (x_p - x)/dx$$

$$w_y = (y_p - y)/dy$$

5.2 Non-standard grids

The interpolation code requires that model data points have a uniform spacing. Rotated lat-lon grids, i.e., those at uniform spacing defined with a rotated pole can be used but stretched grids are not catered for in this release. If the grid cannot be interpolated then the simulation is done at model data point positions. The code will issue a warning message if interpolation has been requested but is not available for that grid.

5.2.1 Coordinate rotation

If a grid has a rotated pole, observation data coordinates (which are assumed to have standard lat-lon coordinates) are transformed to the rotated grid before interpolation. Model data, interpolated or not as appropriate, are then transformed to standard coordinates. The rotation transformation is described below.

Using a left-handed coordinate system with the polar axis as the z-axis, the forward transformation (standard->rotated coordinates) is effectively a sequence of two axis rotations, a z-axis rotation or longitude transformation, such that the rotated pole then lies on the 180.0 degree meridian and then a y-axis rotation or latitude transformation. To convert from rotated to standard coordinates, the transformation, including the rotation sequence, is reversed.

If the rotated pole is at position (λ_p, φ_p) then the rotation angles are

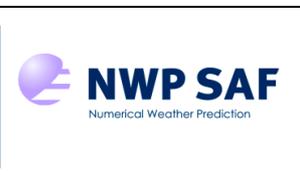
$$\theta_z = \pm(\lambda_p + \pi)$$

$$\theta_y = \pm(\pi/2 - \varphi_p)$$

+ve for the forward transformation, -ve for the reverse.

The z-axis rotation is then:

$$\lambda_z = \lambda - \theta_z$$

		Radiance Simulator v1.0 User Guide	Doc ID : NWPSAF-MO-DS-033 Version : 1.0 Date : 17/11/2014
--	--	---	---

The y-axis rotation:

$$\varphi_y = \sin^{-1}(\sin \varphi \cos \theta_y - \cos \lambda \cos \varphi \sin \theta_y)$$

$$\lambda_y = d \cos^{-1}(\cos \lambda \cos \varphi \cos \theta_y + \sin \varphi \sin \theta_y)$$

Note: The input coordinates (λ , φ) to each rotation change meaning depending on the order of the rotation sequence, e.g., if the z-axis rotation is done first then λ is the original longitude coordinate for the z-axis rotation and then $\lambda = \lambda_z$ for the y-axis rotation.

5.3 Field conversions

Some fields that are required to run RTTOV may not be available directly but can be derived from other fields. The Radiance Simulator will convert (if appropriate) potential temperature to temperature, relative to specific humidity and dewpoint temperature to specific humidity.

Conversion from model units (e.g., kg/kg mixing ratios) to RTTOV units (often PPMV) is done on a per-profile basis for each RTTOV call. RTTOV also requires profiles to be ordered top-down. This is contrary to the usual convention for model data profiles which is to run from the bottom-up but both vertical axis orientations have been allowed for; a check is performed on the first profile to determine the orientation on the assumption that pressure is monotonically increasing in height and the profile is flipped over if necessary. These per-profile conversions are discarded with each iteration, the data available at the end of processing, and which may be written to the output file, is in model units and in the original orientation.

5.3.1 Potential temperature conversion

Temperature T is obtained from potential temperature θ using the standard formula

$$T = \theta \left(\frac{p}{p_0} \right)^\kappa$$

Where:

p = pressure

p_0 = reference pressure (=1000 hPa)

$\kappa = R/c_p$ (R is the specific gas constant for dry air, c_p is the specific heat capacity)

5.3.2 Humidity conversion

Surface (2m) humidity information may be stored as dewpoint temperature T_d or relative humidity. In both cases, in order to obtain specific humidity q it is necessary to calculate specific humidity at saturation.

Specific humidity q (units kg/kg) is defined in terms of pressure as

$$q = \frac{\varepsilon e}{(p - (1 - \varepsilon)e)}$$

Where:

$\varepsilon = R_d/R_v$ (ratio specific gas constant of dry air and water vapour)

e = vapour pressure

p = atmospheric pressure

The saturation vapour pressure e_s is calculated using the commonly used Magnus Formula

$$e_s = a \exp\left(\frac{bT}{c + T}\right)$$

Where T is temperature and a , b , c are empirically derived constants with different values for liquid water and ice. It is assumed that if $T < 273.15$ K then ice coefficients should be used.

For the dewpoint conversion we can substitute $T = T_d$ and $e = e_s$ in the above equations and calculate q .

Relative humidity r is defined as e/e_s but to a very good approximation $r = q/q_s$, therefore q is simply

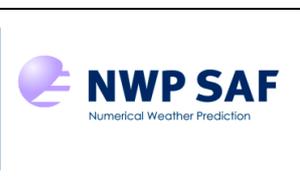
$$q = q_s r$$

Substituting $e = e_s$ in the first equation allows q_s to be calculated.

5.4 Pressure levels

All level-dependent fields are defined on the same set of pressure levels for each profile. The Radiance Simulator will use input pressure level fields if available but usually these have to be calculated from other data. This is almost always the case for any of the currently supported data sources. If the data source is a Met Office UM fieldsfile or an ECMWF GRIB file, sets of coefficients are provided within the file in each case. Users should refer to the relevant documentation for the exact methods. For the NWP SAF diverse profile datasets which originate from the ECMWF model, the coefficients are contained within the code.

If running the MW scattering code, RTTOV requires, in addition, a set of *half-level* pressures and cloud profile fields are defined on layers bounded by these half-levels. In the case of ECMWF pressure levels, half-levels are computed as a by-product of calculating the full levels and retained for use with RTTOV-Scatt. UM *theta* levels are obtained by interpolation from intermediate *rho* levels. The *rho* levels are used as the half-levels in RTTOV-Scatt in this case.

		Radiance Simulator v1.0 User Guide	Doc ID : NWPSAF-MO-DS-033 Version : 1.0 Date : 17/11/2014
--	--	---	---

5.5 Surface type

RTTOV requires the underlying surface to be classified as either *land*, *sea* or *ice*. All input profile datasets should contain a land-sea mask (LSM) either explicitly or by proxy. If there is no LSM but there is an orography field then the surface is classified as *sea* if the surface height is 0.

Some datasets may use a floating point definition of the LSM to accommodate mixed surface types in a grid-box near coastlines. This is, in effect, a land fraction such that 0=*sea only* and 1=*land only*. In the Radiance Simulator it is assumed that such points are *sea* if the LSM value < 0.5, otherwise *land*.

Ideally, the model dataset should also include a sea-ice fraction field which can be used to distinguish between *sea* and *ice*, however this is not compulsory as it may not always be available. If sea-ice fraction is present, then the surface is reclassified from *sea* to *ice* if the fraction of ice > 0.5 but this value can be changed by the user in the configuration namelist.

5.6 Emissivity values

Emissivities are provided by RTTOV in all cases by use of the *calcemis=.true.* option in the RTTOV call - users should consult the RTTOV User Guide for details. Other options may be added in later releases.

5.7 Running RTTOV

Standard clear-sky and IR scattering simulations use the *rttov_direct* code, but if Jacobian or transmittance output has been requested then *rttov_k* is run instead and this will increase the run time. MW scattering simulations use *rttov_scatt*. There is currently no option to produce Jacobians from the MW scattering code as the Radiance Simulator has not been set up to run the adjoint version of RTTOV-Scatt.

Each of the two scattering options requires an additional set of coefficients to be read in prior to running RTTOV (see section 4.1 for specifying these). In all cases, the standard unformatted .dat files are used.

Some small performance gains may be obtained by varying the number of profiles per call to RTTOV. In this release the number has been fixed at 5 per call. In future releases this will be made a tunable parameter.

5.8 Quality control

Quality control flags are set in the *qcflags* variable which is written to the output file. If *qcflags* has a value of zero then there were no errors, otherwise one or more of the following flags will have been set (flag bit numbers here correspond to those defined in the Fortran90 standard for the intrinsic bit manipulation procedures):

Flag/Bit number	Description
0	Observation is outside the model domain
1	Error reported by RTTOV

If the observation location is outside the model domain then no simulation will be performed. Output values for radiances or brightness temperatures will be zero in this case (this is the value used to initialise these variables in RTTOV).

If there is an error reported by RTTOV, the error code is stored in the *qcinfo* variable which is also written to the output file. Users should consult the RTTOV User Guide to determine what the error codes mean. In this case, brightness temperatures may have been calculated but it will depend on the nature of the error.

6. LIMITATIONS AND KNOWN ISSUES

These are documented in the accompanying Release Note. Please report any additional problems via the NWP SAF helpdesk feedback form at <http://nwpsaf.eu/feedback.html>.