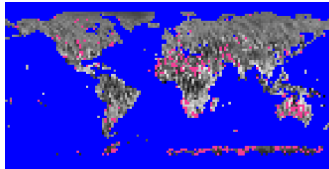


Documentation
SYNTHECLOUD



Synthèses mensuelles d'évènements nuageux

	Nom et Fonction	Date	Signature
Préparé par	N. PASCAL		
Approuvé par	CGTD		
Pour Application			

	Centre de Gestion et de Traitement de Données	Réf: 0512006-NT-UDEV-V01-R03 Rév. : 1.3 Date : 6/03/2006 Page: 2/30
--	----------------------------------------------------------	------------------------------------------------------------------------------------------

DIFFUSION

DIFFUSION INTERNE CGTD	DIFFUSION EXTERNE
--------------------------------------	--------------------------

	Centre de Gestion et de Traitement de Données	Réf: 0512006-NT-UDEV-V01-R03 Rév. : 1.3 Date : 6/03/2006 Page: 3/30
--	----------------------------------------------------------	------------------------------------------------------------------------------------------

<i>Historique et révisions</i>	
02/12/05	v1.0 : Création
12/12/05	v1.1 : Modifications après relecture par J. Descloîtres
17/02/06	V1.2 : Modifications après relecture par A. Lifermann : - Ajout de détails sur les fichiers d'entrée : description, nomenclature, résolution , horodatage. - Mis à jour des options de ligne de commande
6/03/2006	v1.3 : Réordonnancement et ajout de nouvelles variables dans le tableau du paragraphe 3.3

Table des matières

1	Présentation.....	5
1.1	Le projet.....	5
1.2	Le logiciel.....	6
1.3	Fichiers de sortie.....	8
2	Installation.....	9
2.1	UNIX/LINUX.....	9
2.1.1	Dépendances.....	9
2.1.2	Procédure.....	9
3	Extraction des variables.....	10
3.1	Fichiers d'entrée.....	10
3.1.1	Description.....	10
3.1.2	Horodatage.....	11
3.2	Colocalisation des points entre les fichiers.....	12
3.2.1	avec les fichiers IIR.....	12
3.2.2	avec les fichiers METEO.....	13
3.2.3	avec les fichiers MODIS.....	13
3.2.4	avec les fichiers CERES.....	14
3.3	Construction des variables.....	15
4	Structure des fichiers de sortie.....	19
4.1	Variable par variable	19
4.2	Pour une variable donnée.....	20
5	Réalisation Logicielle.....	22
5.1	Déroulement de l'algorithme.....	22
5.2	Architecture logicielle.....	23
5.2.1	Modules de lecture.....	24
5.2.1.1	Interface commune.....	24
5.2.1.2	Spécialisation.....	25
5.2.2	Modules de stockage.....	26
5.2.2.1	Principe.....	26
5.2.2.2	Spécialisation.....	27
5.2.3	Modules d'écriture des fichiers de sortie.....	27
6	Dictionnaire.....	28
7	Annexes.....	29
7.1	Calcul de l'humidité relative à la glace $HR_i=f(HR,T)$	29
7.2	Test inclusion d'un point dans un GRing.....	30

	Centre de Gestion et de Traitement de Données	Réf: 0512006-NT-UDEV-V01-R03 Rév. : 1.3 Date : 6/03/2006 Page: 5/30
--	----------------------------------------------------------	------------------------------------------------------------------------------------------

1 Présentation

1.1 Le projet

Ce projet a pour but de réaliser une synthèse mensuelle d'événements nuageux.

Divers traitements statistiques sont appliqués à des grandeurs physiques lues lors d'événements nuageux détectés par le LIDAR de CALIPSO (CALIOP). Ces résultats (moyenne, pour certaines écart-type) sont intégrés à la fois géographiquement dans des cellules de 2.5°x2.5° et temporellement sur un mois. Le nombre d'évènements dans différentes plages de valeurs de ces grandeurs est ensuite séquencer en fonction d'une variable référence qui peut être :

- La température de sommet de nuage,
- L'indicateur dynamique grande échelle (le vent vertical à 500hPa).

Le résultat de ces synthèses sera écrit dans 4 fichiers HDF :

- un pour les évènements de jour ayant lieu au-dessus de la terre,
- un pour ceux de jour au-dessus de la mer,
- un de nuit au-dessus de la terre,
- et un de nuit au-dessus de la mer.

Rem : La définition jour/nuit est celle utilisée par CALIPSO.

1.2 Le logiciel

Le logiciel est appelé en ligne de commande avec la syntaxe suivante :

```
usage: synthecloud.exe [options]
options:
  -h, --help                show this help message and exit
  -O OUTPUT_DIR, --output=OUTPUT_DIR
                           The output files directory
  -R RADIX, --radix=RADIX
                           The output files radix
  -C CALIOP_DIR, --caliop=CALIOP_DIR
                           The CALIOP files (CAL_LID_L2_CAL_01kmCLay) directory
  -I IIR_DIR, --iir=IIR_DIR
                           The IIR files (CAL_IIR_L2_Track) directory
  -M MODIS_MYD06_DIR, --modis_MYD06=MODIS_MYD06_DIR
                           The MODIS (MYD06) files directory
  -N MODIS_MYD021KM_DIR, --modis_MYD021KM=MODIS_MYD021KM_DIR
                           The MODIS (MYD021KM) files directory
  -G CERES_DIR, --ceres=CERES_DIR
                           The CERES (CER_CRS_Aqua-FM4-MODIS) files directory
  -S CLOUDSAT_DIR, --cloudsat=CLOUDSAT_DIR
                           The CLOUDSAT files directory
  -W METEO_MOM_P_DIR, --mom_p=METEO_MOM_P_DIR
                           The GMAO's METEO tavg3d_mom_p files directory
  -X METEO_MIS_X_DIR, --mis_x=METEO_MIS_X_DIR
                           The GMAO's METEO tsyn2d_mis_x files directory
  -Y METEO_MIS_P_DIR, --mis_p=METEO_MIS_P_DIR
                           The GMAO's METEO tsyn3d_mis_p files directory
  -m MONTH, --month=MONTH
                           The month of the synthesis [1-12]
  -y YEAR, --year=YEAR     The year of the synthesis (YYYY)
  -t MAIN_VAR, --main_var=MAIN_VAR
                           The variable used as main (TEMPERATURE or OMEGA)
  -c, --compress          Enable the HDF compression mode for the output hdf
                           files
  -d, --day               Build the day output files
  -n, --night             Build the night output files
  -f, --force             Force the radix to be used as the output file name
  -s, --skip_mode         If one variable read in a file is undefined,
```

	Centre de Gestion et de Traitement de Données	Réf: 0512006-NT-UDEV-V01-R03 Rév. : 1.3 Date : 6/03/2006 Page: 7/30
--	----------------------------------------------------------	------------------------------------------------------------------------------------------

-v, --verbose	invalidate all variables read in it Print out processing informations
---------------	--------------------------------------------------------------------------

Quelques précisions sur les options :

- *main_var* : quelle variable sera considérée comme la variable référence. Peut être "TEMPERATURE" (par défaut) ou "OMEGA".
- *radix* : le radical par lequel doivent commencer le nom des fichiers de sortie. Par défaut, « synthecloud_ »
- *force* : force le radical a être utilisé comme le nom du fichier de sortie.
- *skip_mode* : permet de choisir le mode de traitement des valeurs manquantes ou erronées pour une variable. Si elle est fixée à :

true : lors de la lecture des variables en un point (lat,lon,time) dans un fichier, si l'une d'entre elles est manquante, toutes les variables lues dans ce fichier seront considérées comme fausses. Elles ne seront alors pas prises en compte.

False (défaut) : seules les variables manquantes ou erronées sont ignorées.

Sur un exemple :

Un événement nuageux a été détecté au point (13,5;98,3) par le LIDAR CALIOP. Toutes les variables pouvant être extraites des fichiers CALIOP et METEO sont bonnes et seront donc prises en compte. Par contre, lors de la lecture du fichier MODIS MYD06_L2, les variables LWC, OPTICAL_THICKNESS, BRIGHTNESS_TEMPERATURE_DIFFERENCE sont valides, mais PARTICLE_SIZE est absente. si le mode est true, les 4 variables seront rejetées. En mode false, seule PARTICLE_SIZE sera rejetée.

	Centre de Gestion et de Traitement de Données	Réf: 0512006-NT-UDEV-V01-R03 Rév. : 1.3 Date : 6/03/2006 Page: 8/30
--	----------------------------------------------------------	------------------------------------------------------------------------------------------

1.3 Fichiers de sortie

Les fichiers de sorties se présentent sous forme de fichiers HDF. Pour plus de détails sur ce format, vous pouvez vous référer à la documentation « tutorial_utilisation_HDF.doc » ou sur le [site officiel](#).

La nomenclature pour le nom des fichiers se présente ainsi :

RADIX_YYYY:MM_DayNightMode_LandWaterMode.hdf

avec :

- RADIX : un préfixe configurable dans l'application,
- YYYY : l'année de la synthèse,
- MM : le mois de la synthèse,
- DayNightMode : « day » ou « night »,
- LandWaterMode : « land » ou « water ».

Rem : On peut aussi fixer le nom de sortie avec l'option *--force* et *--radix*.

	Centre de Gestion et de Traitement de Données	Réf: 0512006-NT-UDEV-V01-R03 Rév. : 1.3 Date : 6/03/2006 Page: 9/30
--	----------------------------------------------------------	------------------------------------------------------------------------------------------

2 Installation

2.1 UNIX/LINUX

2.1.1 Dépendances

Pour être compilé et fonctionner correctement, les bibliothèques suivantes sont nécessaires :

- python >= 2.3
- HDF4.2r1

2.1.2 Procédure

C'est une installation LINUX standard :

1- Dans le répertoire de téléchargement

`tar xzf syntheccloud-x.x.tar.gz`

2- Dans le répertoire d'extraction

`./configure`

`make`

3- Le programme exécutable est alors accessible en :

`{répertoire_installation}/syntheccloud.exe`

	Centre de Gestion et de Traitement de Données	Réf: 0512006-NT-UDEV-V01-R03 Rév. : 1.3 Date : 6/03/2006 Page: 10/30
--	----------------------------------------------------------	-------------------------------------------------------------------------------------------

3 Extraction Des Variables

3.1 Fichiers d'entrée

3.1.1 Description

Fichier	Nom	Provider	Résolution (lat*lon)	Fréquence
CALIOP	CAL_LID_L2_01kmCLay		(1*1)km	1/demi-orbite
IIR	CAL_IIR_L2_Track		(1*1)km	1/demi-orbite
METEO2D	tsyn2d_mis_x	GMAO	(1*1.25)km ; 6hrs	1/jour
METEO3D	tsyn3d_mis_p	GMAO	(1*1.25)km ; 6hrs	1/jour
METEO3D_OMEGA	tavg3d_mom_p	GMAO	(1*1.25)km ; 6hrs	1/jour
MODIS_cloud	MYD06_L2	DAAC	(5*5)km	1/5min
MODIS_radiance	MYD021KM	DAAC	(1*1)km	1/5min
CERES	CER_CRS_Aqua-FM4-MODIS	LaRC	Résolution primaire du capteur CERES	1/hr
CLOUDSAT		CLOUDSAT Data Processing Center		

Centre de Gestion et de Traitement de Données	Réf: 0512006-NT-UDEV-V01-R03 Rév. : 1.3 Date : 6/03/2006 Page: 11/30
----------------------------------------------------------	-------------------------------------------------------------------------------------------

3.1.2 Horodatage

L'horodatage des fichiers est déterminé à partir de leur nom.

Fichier	Nomenclature
CALIOP	CAL_LID_L2_01kmCLay-BETA-V01.YYYY-MM-DDThh-mm-ssZd.hdf
IIR	CAL_IIR_L2_Track-BETA-V01.YYYY-MM-DDThh-mm-ssZd.hdf
METEO2D	DAS.llk.asm.tsyn2d_mis_x.GEOS402.YYYYMMDD00.YYYYMMDD21.V01
METEO3D	DAS.llk.asm.tsyn3d_mis_p.GEOS402.YYYYMMDD00.YYYYMMDD18.V01
METEO3D_OME GA	DAS.llk.asm.tavg3d_mom_p.GEOS402.YYYYMMDD00.YYYYMMDD18.V01
MODIS_cloud	MYD06_L2.AYYYYSSS.hhmm.004.2004078003614.hdf
MODIS_radiance	MYD021KM.AYYYYSSS.hhmm.004.2004022213705.hdf
CERES	CER_CRS_Aqua-FM4-MODIS_Beta1_018025.YYYYMMDDhh
CLOUDSAT	

avec :

- **YYYY** : l'année,
- **MM** : le mois, **DD** : le jour ou **SSS** : le nombre de jours écoulés depuis le début de l'année,
- **d** : si le fichier est de jour (D) ou de nuit (N),
- **hh** : l'heure, **mm** : les minutes, **ss** : les secondes.

	Centre de Gestion et de Traitement de Données	Réf: 0512006-NT-UDEV-V01-R03 Rév. : 1.3 Date : 6/03/2006 Page: 12/30
--	----------------------------------------------------------	-------------------------------------------------------------------------------------------

3.2 Colocalisation des points entre les fichiers

Les évènements nuageux sont déterminés à partir des fichiers CALIOP. Il s'agit donc de déterminer les points coïncidents avec cette mesure dans les fichiers IIR, MODIS, CERES...

Pour cela, il faut déterminer :

- dans quel fichier se trouve le point coïncident,
- rechercher le point le plus proche dans le fichier, et récupérer ses index afin de pouvoir lire la variable dans tous les sds du fichier.

La démarche choisie va dépendre du type de fichier.

3.2.1 avec les fichiers IIR

➤ Recherche du fichier

Le fichier IIR contenant le point coïncident possède le même horodatage que le fichier CALIOP. Cette date est déterminé à partir du nom du fichier.

Ainsi, pour une mesure provenant du fichier

« CAL_LID_L2_01kmCLay-BETA-V01.2004-01-01T00-20-11ZN.hdf »,

le point coïncident sera présent dans le fichier

« CAL_IIR_L2_Track-BETA-V01.2004-01-01T00-20-11ZN.hdf »

➤ Recherche du point

On recherche le point possédant un *LIDAR_shot_time* égal au *time* du tir LIDAR source.

	Centre de Gestion et de Traitement de Données	Réf: 0512006-NT-UDEV-V01-R03 Rév. : 1.3 Date : 6/03/2006 Page: 13/30
--	----------------------------------------------------------	-------------------------------------------------------------------------------------------

3.2.2 avec les fichiers METEO

➤ Recherche du fichier

Il y a un fichier météo par jour, couvrant la surface du globe. Il suffit donc rechercher le fichier du jour correspondant au tir LIDAR. La date est déterminé à partir du nom du fichier.

➤ Recherche du point

Les fichiers météo sont des fichiers hdf, avec, par sds :

- une dimension pour la latitude avec un écart de 1° entre 2 données,
- une pour la longitude avec un écart delta_lon de 1,25° entre 2 données,
- une pour l'heure avec un écart delta_time dépendant du type de fichier, mais constant.

Ces écarts étant constants, le point coïncident peut être déterminé avec un index par dimension calculé ainsi :

$$\text{lat_index} = (\text{lat} - \text{lat_min}) / \text{delta_lat} ,$$

$$\text{lon_index} = (\text{lon} - \text{lon_min}) / \text{delta_lon} ,$$

$$\text{time_index} = (\text{time} - \text{time_min}) / \text{delta_time} ,$$

où lat_min, lon_min et time_min sont les valeurs minimales du fichier météo pour chaque dimension.

3.2.3 avec les fichiers MODIS

➤ Recherche du fichier

Les satellites portant MODIS et CALIOP ont une orbite proche, et se suivent avec un écart de ~75s, MODIS en tête. Un point CALIOP pourra ainsi avoir une coïncidence dans le fichier MODIS de même horodatage, déterminé grâce à son nom, ou son prédécesseur. En pratique,

- si le tir LIDAR est à plus de 120 sec du fichier MODIS de même heure, la coïncidence est dans ce fichier,
- à moins de 120 sec mais plus de 60 (bornes comprises), on teste les deux fichiers pour savoir lequel contient le point,
- à moins de 60, le point est dans le prédécesseur.
- Entre 60 et 120 sec, pour savoir quel fichier contient la coïncidence, on va rechercher le point le plus proche du point CALIOP dans les 2 fichiers, et prendre le fichier contenant le plus proche.

➤ Recherche du point

Le recherche des index du point utilise une méthode de force brute : on va parcourir les points des colonnes du dernier quart de granule et rechercher le plus proche voisin.

	Centre de Gestion et de Traitement de Données	Réf: 0512006-NT-UDEV-V01-R03 Rév. : 1.3 Date : 6/03/2006 Page: 14/30
--	----------------------------------------------------------	-------------------------------------------------------------------------------------------

3.2.4 avec les fichiers CERES

➤ Recherche du fichier

Le principe est le même que pour MODIS (même plateforme, même type de fichiers). La différence est la couverture temporelle, d'une heure dans le cas des fichiers CERES. Avec une telle couverture temporelle, le point coïncident est quasiment toujours dans le fichier contenant l'heure du point CALIOP.

➤ Recherche du point

On recherche les index du point le plus proche de la même façon que pour MODIS.

3.3 Construction des variables

Variable	Fichier source	Sds utilisé	Etape Intermédiaire ou Résultat
Température vraie de sommet de nuage <i>Tc</i>	CALIOP (<i>lat,lon</i>)	Layer_Top_Altitude	Altitude de sommet de nuage <i>ct_height</i>
		Profile_Time	Moment du shot LIDAR <i>time</i>
	METEO3D (<i>lat,lon,time,ct_height</i>)	TMPU	Température <i>Tc</i>
Température Tropopause <i>Ttropo</i>	CALIOP (<i>lat,lon</i>)	Tropopause_Temperature	<i>Ttropo</i>
Température de la base du nuage <i>Tcb</i>	CALIOP (<i>lat,lon</i>)	Layer_Top_Altitude	Altitude de la base du nuage <i>cb_height</i>
		Profile_Time	Moment du shot LIDAR <i>time</i>
	METEO3D (<i>lat,lon,time,cb_height</i>)	TMPU	Température <i>Tcb</i>
Température de surface <i>Ts</i>	METEO2D (<i>lat,lon,time</i>)	TSKIN	<i>Ts</i>
Température à 2m sous la surface <i>T2m</i>	METEO2D (<i>lat,lon,time</i>)	T2M	<i>T2m</i>
Température à 10m sous la surface <i>T10m</i>	METEO2D (<i>lat,lon,time</i>)	T10M	<i>T10m</i>
Stabilité Statique <i>StatStab</i>	METEO3D (<i>lat,lon,time,700hPa</i>)	TMPU	Température <i>T_700hPa</i>
	METEO3D (<i>lat,lon,time,1000hPa</i>)	TMPU	Température <i>T_1000hPa</i>
	<i>StatStab=T_700hPa-T_1000hPa</i>		
Température Brillance à 12µm <i>BT12</i>	IIR(<i>lat,lon</i>)	Brightness_Temperature_12_05	<i>BT12</i>
Différence de température de brillance <i>DBT</i>	MODIS(<i>lat,lon</i>)	Brightness_Temperature_Difference	<i>DBT</i>

**Centre de Gestion et de Traitement
de Données**

Réf:

0512006-NT-UDEV-V01-R03

Rév. : 1.3 Date : 6/03/2006

Page: 16/30

Pression Sommet 1ère couche nuageuse <i>Pct</i>	CALIOP (<i>lat,lon</i>)	Layer_Top_Altitude	Altitude de sommet de nuage <i>ct_height</i>
		Profile_Time	Moment du shot LIDAR <i>time</i>
	METEO3D (<i>lat,lon,time,ct_height</i>)	Default Pressure Levels	Pression <i>Pct</i>
Pression dernière couche nuageuse vue par le LIDAR <i>Pcb</i>	CALIOP (<i>lat,lon</i>)	Layer_Top_Altitude	Altitude dernière couche nuageuse <i>cb_height</i>
		Profile_Time	Moment du shot LIDAR <i>time</i>
	METEO3D (<i>lat,lon,time,cb_height</i>)	Default Pressure Levels	Pression <i>Pcb</i>
Nombre de couches nuageuses <i>Nb_Layer_found</i>	CALIOP (<i>lat,lon</i>)	Number_Layers_Found	<i>Nb_layer_CALIOP</i>
	LOUDSAT(<i>lat,lon</i>)	Pas encore fait	<i>Nb_layer_CLOUDSAT=-1</i> à l'heure actuelle
	$\text{Max} (Nb_layer_CLOUDSAT, Nb_layer_CALIOP) = Nb_Layer_Found$		
Epaisseur Optique <i>OptTh</i>	IIR (<i>lat,lon</i>)	Optical_Depth_12_05	<i>OptTh_IIR</i>
	MODIS(<i>lat,lon</i>)	Cloud_Optical_Thickness	<i>OptTh_MODIS</i>
	$\text{Max} (OptTh_IIR, OptTh_MODIS) = Opt_Th$		
Humidité Relative dans le nuage (au milieu) <i>HR</i>	CALIOP (<i>lat,lon</i>)	Midlayer_Temperature	Température de milieu de nuage <i>Tcm</i>
	METEO3D (<i>lat,lon,time,Tcm</i>)	RH	Humidité relative <i>HR</i>
Humidité dans le nuage (au milieu) par rapport à la glace <i>HRi</i>	CALIOP (<i>lat,lon</i>)	Midlayer_Temperature	Température de milieu de nuage <i>Tcm</i>
	METEO3D (<i>lat,lon,time,Tcm</i>)	RH	Humidité relative <i>HR</i>
	$HRi=f(Hr,Tcm)$; cf annexe 1		
Type nuageux colonne atmosphérique <i>CCT</i>	CALIOP (<i>lat,lon</i>)	Feature_Classification_Flags	<i>CCT</i>
Forme des particules <i>Pshape</i>	IIR(<i>lat,lon</i>)	Particle_Shape_Index	<i>Pshape</i>
Taille des particules <i>Psize</i>	IIR(<i>lat,lon</i>)	Effective_Particle_Size	<i>Psize_IIR</i>
	MODIS(<i>lat,lon</i>)	Effective_Particle_Radius (x2)	<i>Psize_MODIS/GSFC</i>
	CERES(<i>lat,lon</i>)	Mean ice particle effective diameter for cloud layer (3.7)	<i>Psize_MODIS/LaRC</i>

**Centre de Gestion et de Traitement
de Données**

Réf:

0512006-NT-UDEV-V01-R03

Rév. : 1.3 Date : 6/03/2006

Page: 17/30

Liquid Water Content <i>LWC</i>	MODIS(<i>lat,lon</i>)	Water_Path	<i>LWC</i>
Ice Water Content <i>IWC</i>	IIR (<i>lat,lon</i>)	Ice_Water_Path	<i>IWC_IIR</i>
	CERES(<i>lat,lon</i>)	Mean ice water path for cloud layer (3.7)	<i>IWC_MODIS/LaRC</i>
Phase thermodynamique de sommet de nuage <i>IWPh</i>	CALIOP (<i>lat,lon</i>)	Feature_Classification_Flags	<i>IWPh</i>
Dynamique de sommet de nuage <i>u,v,w</i>	CALIOP (<i>lat,lon</i>)	Layer_Top_Altitude	Altitude de sommet de nuage <i>ct_height</i>
		Profile_Time	Moment du shot LIDAR <i>time</i>
	METEO3D (<i>lat,lon,time,ct_height</i>)	UWND	Vent zonal <i>u</i>
	METEO3D_W (<i>lat,lon,time,ct_height</i>)	VWND	Vent méridional <i>v</i>
Indicateur dynamique Grande échelle <i>ID_500</i>	METEO3D_W (<i>lat,lon,time,500hPa</i>)	OMEGA	<i>ID_500</i> (= <i>w(500hPa)</i>)
		OMEGA	<i>ID_800</i> (= <i>w(800hPa)</i>)
Indicateur dynamique Grande échelle <i>ID_800</i>	METEO3D_W (<i>lat,lon,time,800hPa</i>)	OMEGA	<i>ID_800</i> (= <i>w(800hPa)</i>)
		OMEGA	<i>ID_800</i> (= <i>w(800hPa)</i>)
Short Wave Flux <i>FSW</i>	CERES(<i>lat,lon</i>)	CERES SW TOA flux – upwards	<i>FSW</i>
Forçage Short Wave Flux <i>FSW_forcing</i>	CERES(<i>lat,lon</i>)	Shortwave flux - upward – clear	<i>FSW_clear_sky</i>
	CERES(<i>lat,lon</i>)	Shortwave flux - upward - total	<i>FSW_total_sky</i>
	$FSW_total_sky - FSW_clear_sky = FSW_forcing$		
Long Wave Flux <i>FLW</i>	CERES(<i>lat,lon</i>)	CERES LW TOA flux - upwards	<i>FLW</i>
Forçage Long Wave Flux <i>FLW_forcing</i>	CERES(<i>lat,lon</i>)	Longwave flux - upward - clear	<i>FLW_clear_sky</i>
	CERES(<i>lat,lon</i>)	Longwave flux - upward - total	<i>FLW_total_sky</i>
	$FLW_total_sky - FLW_clear_sky = FLW_forcing$		

	Centre de Gestion et de Traitement de Données	Réf: 0512006-NT-UDEV-V01-R03 Rév. : 1.3 Date : 6/03/2006 Page: 18/30
--	----------------------------------------------------------	-------------------------------------------------------------------------------------------

Indicateur de convection I_{Conv}	?		
Indicateur d'acide Nitrique I_{HNO3}	IIR(<i>lat,lon</i>)	Brightness_Temperature_10_60	<i>BT_10.6</i>
	IIR(<i>lat,lon</i>)	Brightness_Temperature_12_05	<i>BT_12.5</i>
	$I_{HNO3}=BT_{10.6}-BT_{12.5}$		
Indicateur WV I_{WV}	MODIS_RADIANCE (<i>lat,lon</i>)	EV_1KM_RefSB, idx 12	<i>Radiance_channel_18</i>
	MODIS_RADIANCE (<i>lat,lon</i>)	EV_250_Aggr1km_RefSB, idx 1	<i>Radiance_channel_2</i>
	$I_{WV}=Radiance_channel_18/Radiance_channel_2$		

4 Structure Des Fichiers De Sortie

Les résultats sont stockés dans les fichiers de sortie Hdf sous forme de sds. Ils sont organisés :

- D'abord variables par variables, en commençant par la variable référence,
- Puis, pour une même variable,
 - un sds contenant la moyenne de la variable dans chaque cellule,
 - puis un sds par classe de la variable de référence.

4.1 Variable par variable

```
SDS list
1: ( 72 x 144 ) AvgTc
2: ( 14 x 72 x 144 ) Tc_Cls
3: ( 72 x 144 ) Avg_Tcb
4: ( 9 x 72 x 144 ) Tcb_Cls_Tc<197
5: ( 9 x 72 x 144 ) Tcb_Cls_Tc=197_200
6: ( 9 x 72 x 144 ) Tcb_Cls_Tc=200_203
7: ( 9 x 72 x 144 ) Tcb_Cls_Tc=203_206
8: ( 9 x 72 x 144 ) Tcb_Cls_Tc=206_209
9: ( 9 x 72 x 144 ) Tcb_Cls_Tc=209_215
10: ( 9 x 72 x 144 ) Tcb_Cls_Tc=215_220
11: ( 9 x 72 x 144 ) Tcb_Cls_Tc=220_230
12: ( 9 x 72 x 144 ) Tcb_Cls_Tc=230_240
13: ( 9 x 72 x 144 ) Tcb_Cls_Tc=240_250
14: ( 9 x 72 x 144 ) Tcb_Cls_Tc=250_260
15: ( 9 x 72 x 144 ) Tcb_Cls_Tc=260_273
16: ( 9 x 72 x 144 ) Tcb_Cls_Tc=273_280
17: ( 9 x 72 x 144 ) Tcb_Cls_Tc>280
18: ( 72 x 144 ) Avg_Ttropo
19: ( 8 x 72 x 144 ) Ttropo_Cls_Tc<197
20: ( 8 x 72 x 144 ) Ttropo_Cls_Tc=197_200
:
```

La **variable référence** (ici Tc, la température de sommet de nuage) se trouve dans les 2 premiers sds, avec :

- sa valeur **moyenne** dans chaque cellule (sds #1)
- le **nombre d'évènements** dans **chaque classe** et dans **chaque cellule** (sds #2)

Les **autres variables** (ici Tcb, la température de base de nuage) dans les sds suivants avec pour chacune :

- sa **moyenne**
- le **nombre d'évènements**, avec **un sds** pour **chaque classe de la variable de référence** (ici Tc).

Chaque sds est lui-même composé de 3 dimensions :
[classe_de_la_variable (ici Tcb), index_latitude, index_longitude]

4.2 Pour une variable donnée

Dans ce paragraphe, le terme « classe » n'a aucun sens informatique. Il signifie *classe de variable*, au sens où il est utilisé dans le sujet du projet (Cf Annexe 2)

Les résultats sont stockés dans les fichiers de sortie sous forme de sds. Ils sont organisés ainsi :

```
SDS list
1: ( 72 x 144 ) AvgTc
2: ( 14 x 72 x 144 ) Tc_Cls
3: ( 72 x 144 ) Avg_Tcb ①
4: ( 9 x 72 x 144 ) Tcb_Cls_Tc<197
5: ( 9 x 72 x 144 ) Tcb_Cls_Tc=197_200
6: ( 9 x 72 x 144 ) Tcb_Cls_Tc=200_203
7: ( 9 x 72 x 144 ) Tcb_Cls_Tc=203_206
8: ( 9 x 72 x 144 ) Tcb_Cls_Tc=206_209
9: ( 9 x 72 x 144 ) Tcb_Cls_Tc=209_215
10: ( 9 x 72 x 144 ) Tcb_Cls_Tc=215_220 ②
11: ( 9 x 72 x 144 ) Tcb_Cls_Tc=220_230
12: ( 9 x 72 x 144 ) Tcb_Cls_Tc=230_240
13: ( 9 x 72 x 144 ) Tcb_Cls_Tc=240_250
14: ( 9 x 72 x 144 ) Tcb_Cls_Tc=250_260
15: ( 9 x 72 x 144 ) Tcb_Cls_Tc=260_273
16: ( 9 x 72 x 144 ) Tcb_Cls_Tc=273_280
17: ( 9 x 72 x 144 ) Tcb_Cls_Tc>280
18: ( 72 x 144 ) Avg_Ttropo
19: ( 8 x 72 x 144 ) Ttropo_Cls_Tc<197
20: ( 8 x 72 x 144 ) Ttropo_Cls_Tc=197_200
    ⋮
```

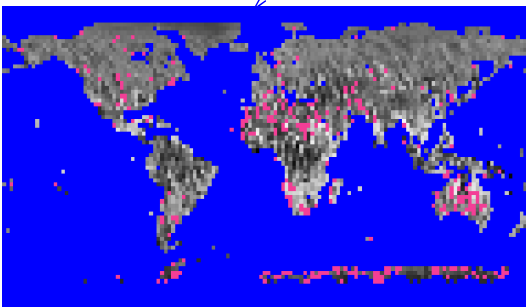
Pour chaque variable (ci-dessus Tcb la température de la base du nuage) sont construits :

- une carte ① de la moyenne de la variable dans les cellules de $2,5^\circ \times 2,5^\circ$,
Rem : Ce sds est absent pour 2 variables : Particle_Shape et Ice_Water_Phase, leur moyenne n'ayant pas vraiment de sens.
- des sds ② à 3 dimensions, contenant les cartes du nombre d'évènements par classe de la variable. On aura un sds de ce type par classe de la variable principale (ici Tc la température de sommet de nuage).

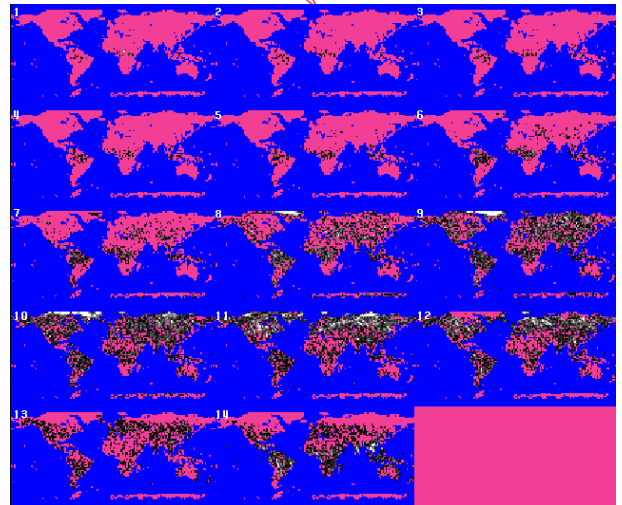
Dans l'exemple ci-dessus, il y aura 14 sds de ce type, car il y a 14 classes de Tc, et chaque sds aura une 3ème dimension de 9 car il y a 9 classes de la variable Tcb. Les dimensions de 72 et 144 correspondent au nombre de cellules de $2,5^\circ \times 2,5^\circ$, respectivement en latitude et longitude.

Exemple sur une variable : Tc la Température Vraie de sommet de nuage

SDS	list
1:	(... 72 x . 144) AvgTc
2:	(14 x 72 x 144) Tc_Cls
3:	(72 x 144) Avg_Tcb
4:	(9 x 72 x 144) Tcb_Cls_Tc<197
5:	(9 x 72 x 144) Tcb_Cls Tc=197 200



Moyenne



Nombre d'évènements dans chaque classe de la variable Tc

Quelques caractéristiques

Nom des sds : la classe de la variable principale apparaît dans le nom des sds dénombrant les événements.

Fill value : les valeurs de remplissage des données absentes ou erronées sont :

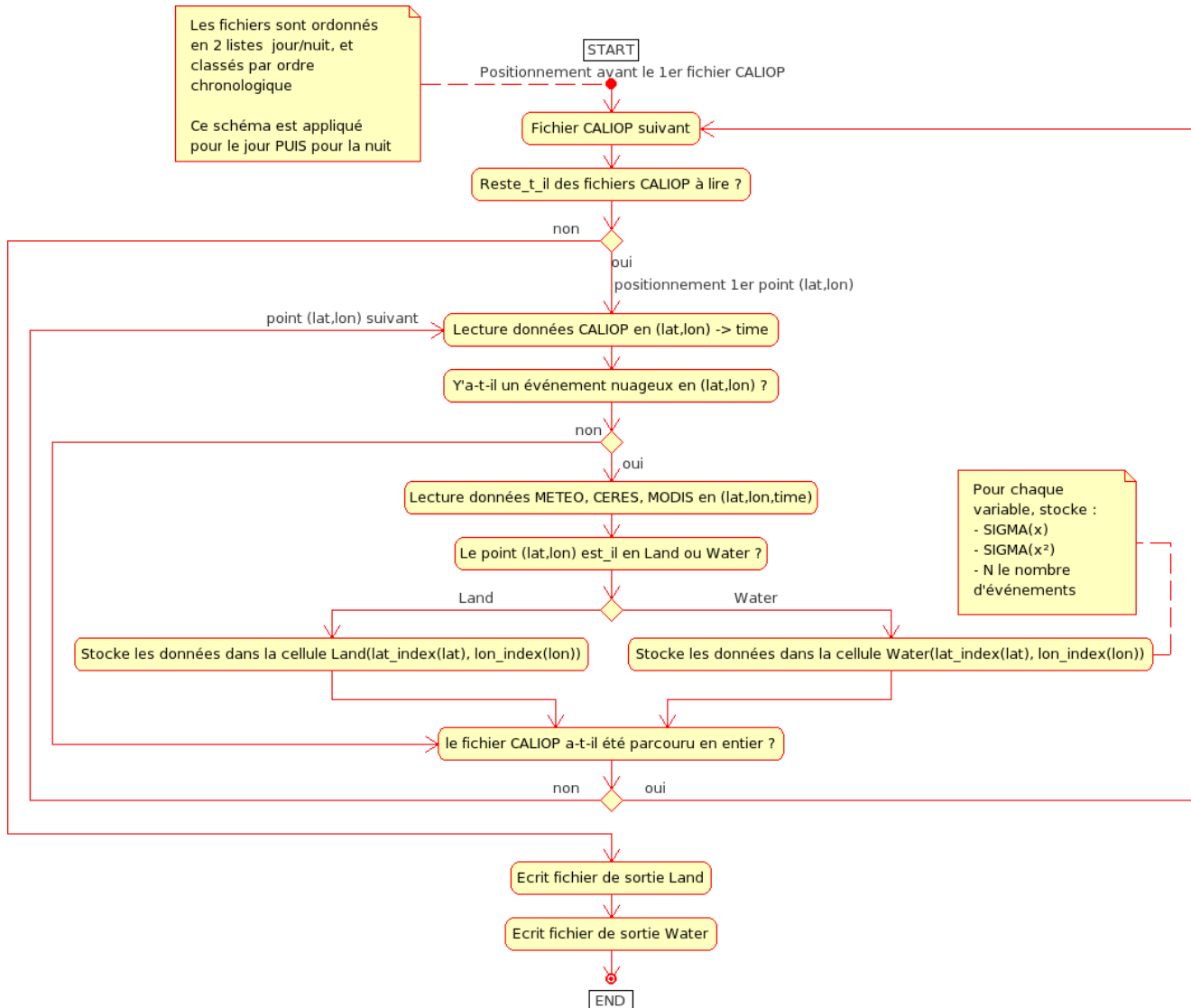
- **-inf** pour les sds typés en float32 (les moyennes). C'est la valeur minimale utilisable pour ce type.
- **65535** pour ceux en uint16 (les nombres d'évènements). C'est la valeur maximale de ce type.

Classes de variable : les limites des classes de variable apparaissent dans les attributs de chaque sds dénombrant les évènements.

Points non nuageux : afin de visualiser la couverture géographique du satellite, une valeur différente (-9999) est donnée aux cellules survolées par le satellite, mais où aucun événement nuageux n'a été détecté, et aux cellules non survolées (*fill value*).

5 Réalisation Logicielle

5.1 Déroulement de l'algorithme



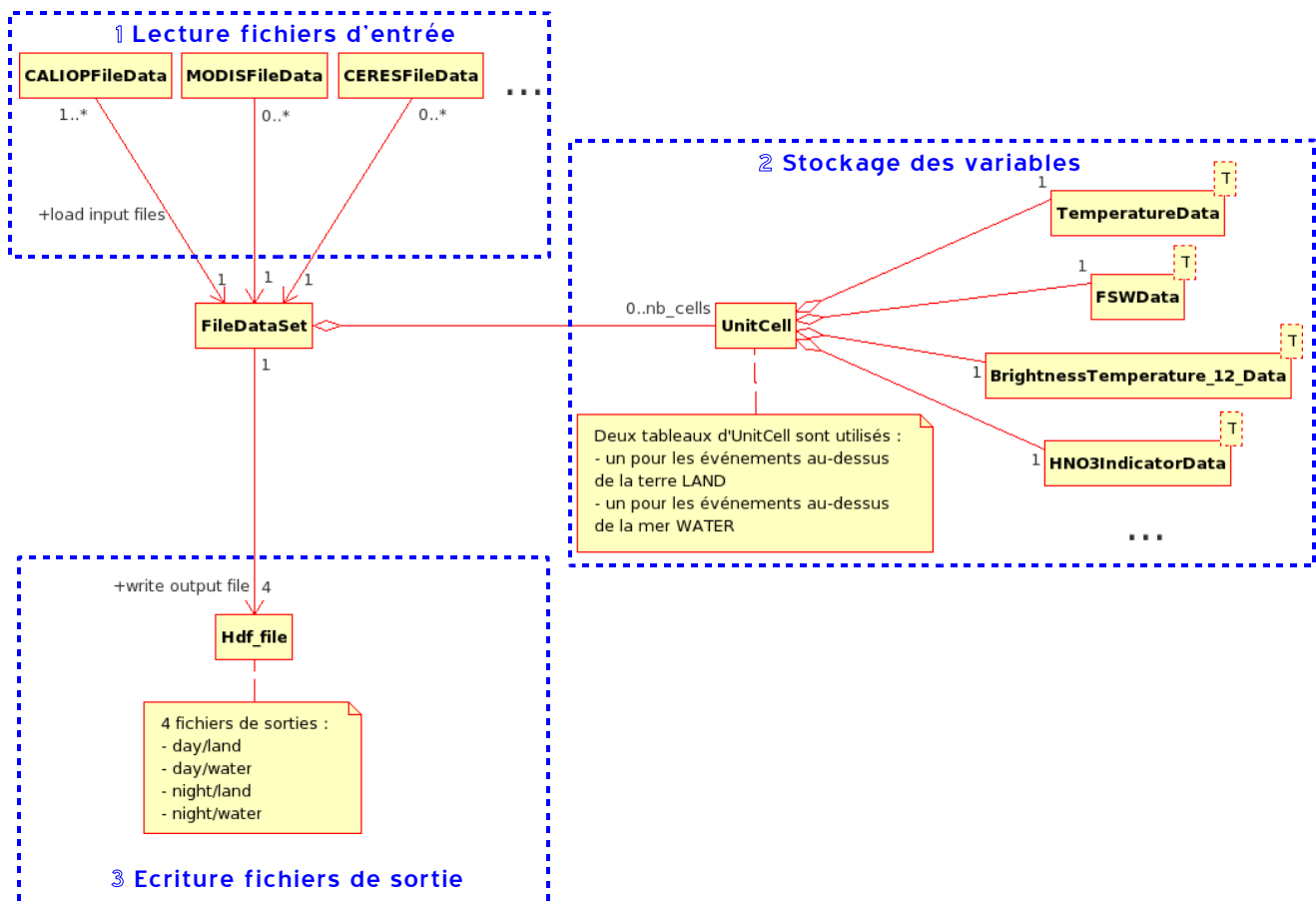
- Les différents fichiers d'entrée sont lus à l'aide des modules XXXFileData (où XXX est le nom du fichier),
- Lorsqu'un événement nuageux est détecté en un point (lat,lon), les valeurs des différentes variables sont lues, et le module FileDataSet en détermine les classes de la variable principale et de la variable elle-même.
- Ces données sont ensuite transmises à la classe UnitCell (une cellule de 2,5°x2,5°) correspondant à (lat,lon).
- Puis, pour chaque variable, la valeur est stockée dans la classe idoine qui met à jour : un accumulateur

- des valeurs, un accumulateur du carré des valeurs et le nombre d'évènement dans chaque classe.
- Enfin, une fois que tous les fichiers ont été parcourus, il ne reste plus qu'à écrire les valeurs des variables contenues dans chaque UnitCell dans les fichiers de sorties.

5.2 Architecture logicielle

La réalisation a été faite en C++ sous LINUX (Mandriva 10.2), et utilise les bibliothèques : hdf pour les entrées-sorties sur les fichiers hdf,

- Hdf_file développée par Fabrice Ducos.



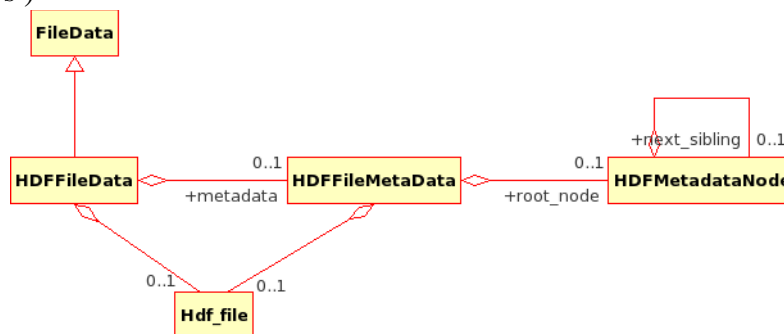
L'architecture se décompose ainsi :

- les modules de lecture XXXFileData (où XXX est le nom du fichier) permettant de lire les différentes variables,
- le module de stockage UnitCell qui stocke pour chaque cellule de 2,5°x2,5° les valeurs des différentes variables (accumulateur des valeur, accumulateur du carré des valeurs, occurrence dans chaque classe de la variable principale et de la variable elle-même) dans les modules correspondants. Il existe un module pour chaque variable, nommé NomDeLaVariableData (où étonnamment NomDeLaVariable est le nom de la variable...),
- L'écriture des fichiers de sortie est faite par l'intermédiaire de la classe Hdf_file,
- le module FileDataSet assure la communication entre ces différents modules.

5.2.1 Modules de lecture

5.2.1.1 Interface commune

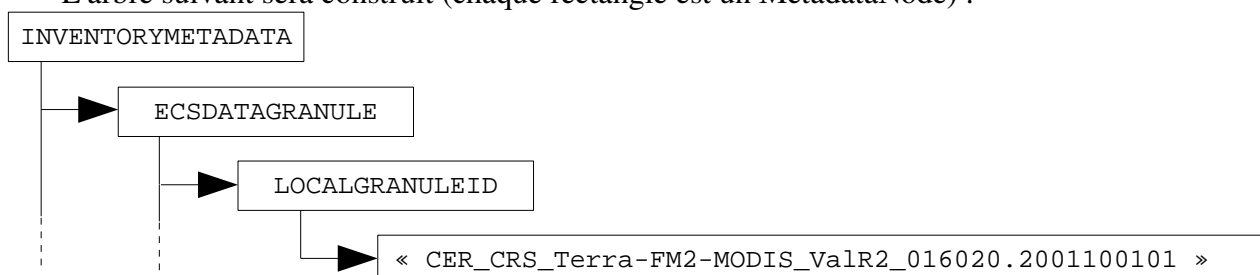
Tous ces modules de lecture utilisent une même interface (mais peuvent néanmoins avoir leurs propres fonctionnalités)



Chaque module de lecture est une spécialisation du module HDFFileData. Il est lui-même composé d'un module Hdf_file qui gère l'accès de bas niveau aux fichiers hdf (lecture/écriture), d'un module HDFFileMetaData qui lit les métadonnées et construit un arbre les représentant. Cet arbre est composé de noeuds HDFMetadataNode. Par exemple, avec les métadonnées suivantes :

GROUP	= INVENTORYMETADATA
GROUPTYPE	= MASTERGROUP
GROUP	= ECSDATAGRANULE
OBJECT	= LOCALGRANULEID
NUM_VAL	= 1
VALUE	= « CER_CRS_Terra-FM2-MODIS_ValR2_016020.2001100101 »
END_OBJECT	= LOCALGRANULEID
END_GROUP	= ECSDATAGRANULE
END_GROUP	= INVENTORYMETADATA

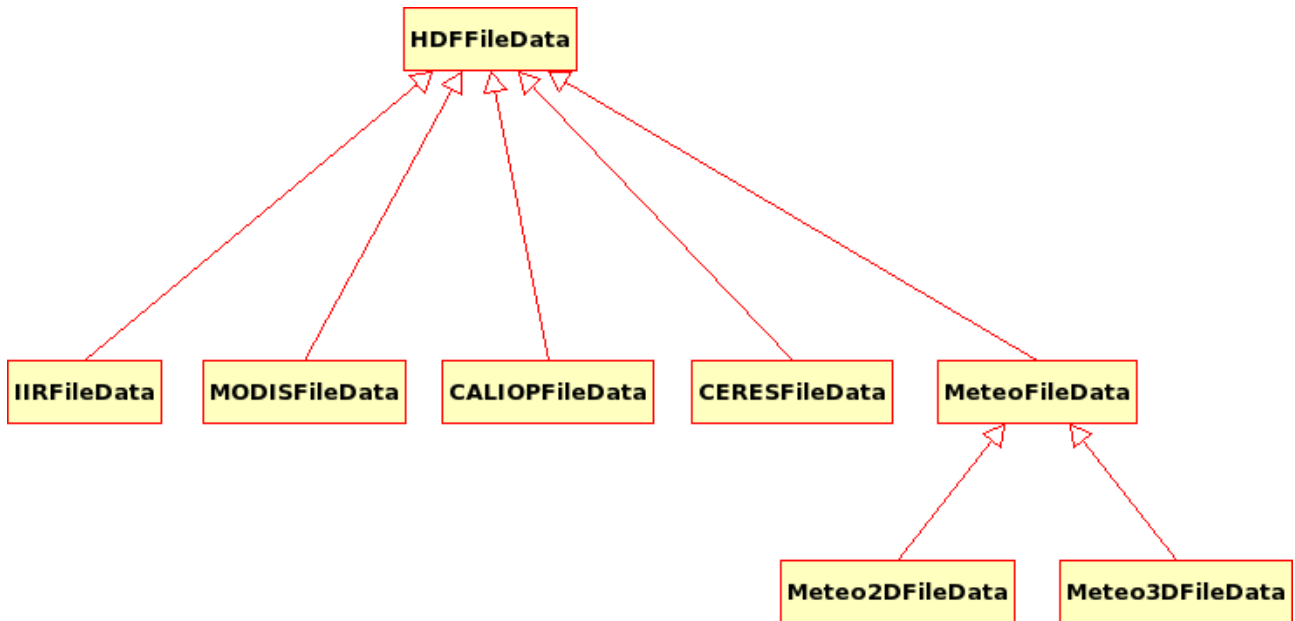
L'arbre suivant sera construit (chaque rectangle est un MetadataNode) :



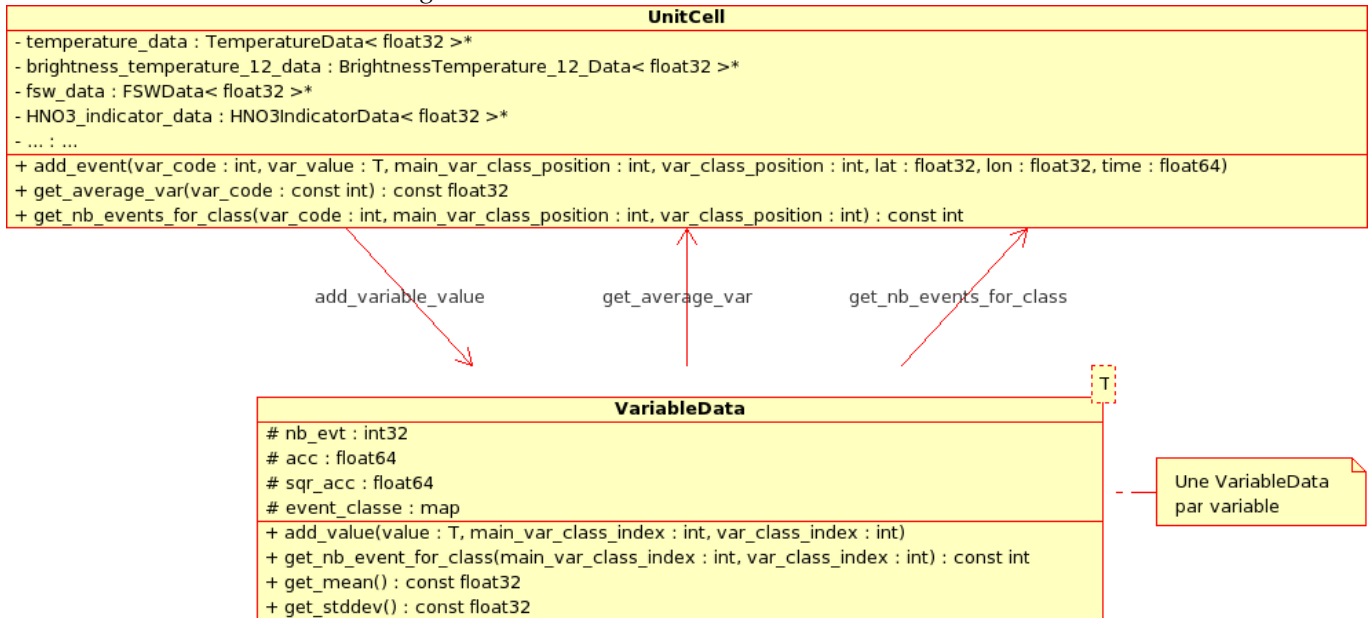
- Le module FileData définit juste une interface abstraite d'accès à des fichiers de données. Elle ne définit pas de fonctionnalité propre.

5.2.1.2 Spécialisation

Les modules de lecture sont hiérarchisés ainsi :



5.2.2 Modules de stockage



5.2.2.1 Principe

Une instance de **UnitCell** représente une cellule de 2,5°x2,5°.

Dans chaque **UnitCell**, une instance spécialisée de **VariableData** par variable (ex : **FSWDData** pour la variable FSW).

Voici comment se déroule la communication entre ces 2 modules sur 2 exemples :

Ajout d'un événement :

- la méthode **UnitCell-> add_event** est appelée, avec :

var_code : un entier représentant la variable à ajouter.

var_value : la valeur de la variable

main_var_class_position : l'index de la classe de variable principale de l'évènement

var_class_position : l'index de la classe de variable de la valeur

- Puis **UnitCell-> add_event** appelle la méthode où VariableXXX est le nom de la variable

correspondant à var_code.

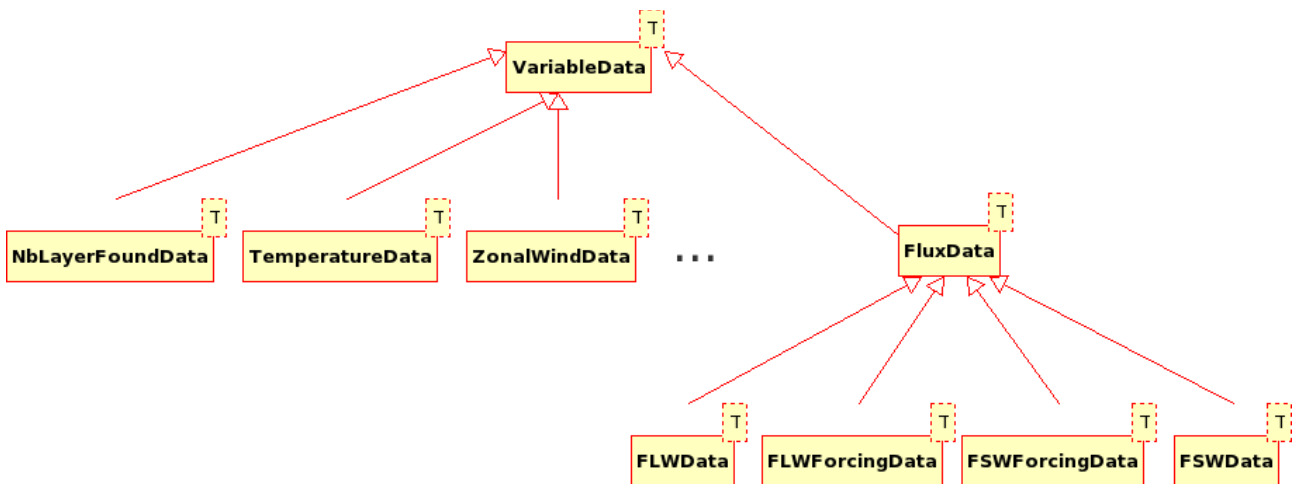
Rem : pour le détail des var_code utilisés, voir la documentation technique de de la classe UnitCell.

Lecture de la moyenne d'une variable :

- la méthode **UnitCell-> get_average_var** appelle la méthode **VariableXXXData->get_mean** correspondante à var_code,
- **VariableXXXData->get_mean** calcule acc/nb_evt et en renvoie la valeur,
- Le même principe est appliqué pour la lecture du nombre d'évènements par classe.

5.2.2.2 Spécialisation

Les classes de stockage de variables **VariableXXXData** sont hiérarchisées comme suit :



5.2.3 *Modules d'écriture des fichiers de sortie*

L'écriture des fichiers de sortie est faite via le module de bas niveau Hdf_file. Voir la documentation de ce module pour plus de détails.

	Centre de Gestion et de Traitement de Données	Réf: 0512006-NT-UDEV-V01-R03 Rév. : 1.3 Date : 6/03/2006 Page: 28/30
--	----------------------------------------------------------	-------------------------------------------------------------------------------------------

6 Dictionnaire

Ci-dessous un dictionnaire de termes utilisés dans cette documentation et dans le projet. Le code étant documenté en anglais, on trouvera entre parenthèses la traduction utilisée pour chaque terme.

Cellule (*cell*) : zone géographique de 2,5°x2,5° dans laquelle sont agrégés les différents événements nuageux.

Classe (*class*) : les valeurs d'une variable sont décomposées en différentes plages appelées classes.

Ex : Si les classes de pression (en hPa) sont :

#1 : <200

#2 : 200<=P<250

#3 : 250<=P<300

#4 : 300<=P<350

#5 : 350<=P

Alors, une pression de 273hPa sera de classe 3

Évènement (*event*) : sous-entendu événement nuageux. C'est un point géographique où un nuage a été détecté par le LIDAR CALIOP

GRing (*GRing*) : définit un polygone entourant au plus près une forme quelconque. La caractéristique est que l'on peut définir le pavage d'une forme quelconque à l'aide de Grings.

Variable (*variable*) : une grandeur physique lue dans n'importe quel type de fichier(s).

Variable indirecte (*indirect variable*) : une variable construite à l'aide de différentes sous-variables issues de différents types de fichiers ou ayant subi un traitement arithmétique.

Variable de référence (*main variable*) : c'est la variable particulière avec laquelle sont classées toutes les autres variables.

7 Annexes

7.1 Calcul de l'humidité relative à la glace $HR_i=f(HR,T)$

Source : Marjolaine Chiriaco

- Par définition, $RH_i = 100 \frac{e}{e_{s,i}}$ (i), où :

- e est la pression partielle (en Pa),
- $e_{s,i}$ est la pression de vapeur saturante par rapport à la glace (en Pa).

- De même, $e = e_{s,w} \frac{RH_w}{100}$ (ii), où :

- RH_w est l'humidité relative à l'eau liquide,
- $e_{s,w}$ est la pression de vapeur saturante par rapport à l'eau liquide (en Pa).

- On a aussi les approximations suivantes pour le calcul des pressions de vapeur saturantes :

$$e_{s,w} = \exp(AT^{-1} + B + CT + DT^2 + E \ln(T)) \quad \text{(iii), où}$$

A=-6096.9385 ; B=21.2409642 ; C=-2.711193E-2 ; D=1.673952E-5 ; E=2.433502

$$e_{s,i} = \exp(AT^{-1} + B + CT + DT^2 + E \ln(T)) \quad \text{(iv), où}$$

A=-6024.5282 ; B=29.32707 ; C=1.0613868E-2 ; D=-1.3198825E-5 ; E=-0.49382577 ;
T la température (en K)

- (ii) dans (i) : $RH_i = \frac{e_{s,w}}{e_{s,i}} RH_w$

Ce qui, tous calculs faits, donne :

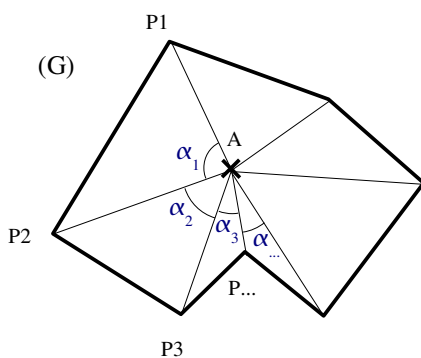
$$RH_i = \exp(AT^{-1} + B + CT + DT^2 + E \ln(T)) RH_w, \text{ avec}$$

A=-72.410300 ; B=- 8.0861058 ; C=- 3.7725800E-2; D=2.9938345E-5 ; E= 2.92732777

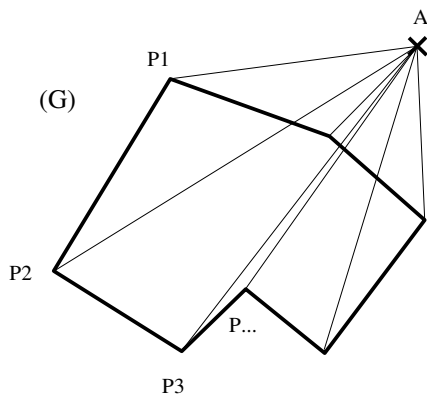
7.2 Test inclusion d'un point dans un GRing

Cette solution a été trouvée sur le site <http://astronomy.swin.edu.au/~pbourke/geometry/insidepoly/>

L'idée est de calculer la somme des angles (orientés et en radians) entre le point à tester et chaque couple de points consécutifs formant le polygone. Si la somme est 2π le point est à l'intérieur, si elle est nulle, il est à l'extérieur. Cette méthode marche aussi pour polygone présentant des "trous".



Si $\sum \alpha_i = 2\pi$ alors A est à l'intérieur du polygone (G)



Si $\sum \alpha_i = 0$ alors A est à l'extérieur du polygone (G)