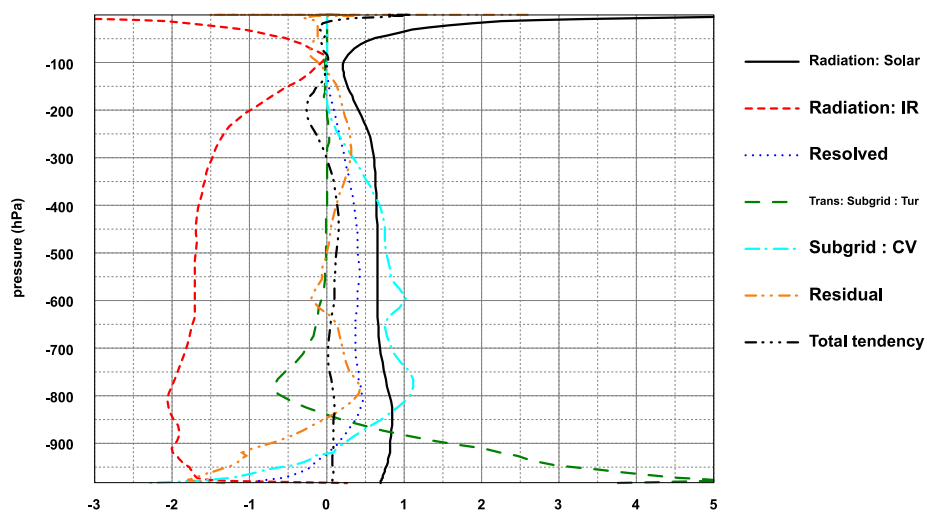


Diagnostics in Horizontal Domains (DDH)

Variables and budget equations, in horizontal mean
ARPEGE, ALADIN and AROME models
Guide for users and developers

9 octobre 2024

TEMPERATURE BUDGET (K/day) , OPER (20221122T0000P.DHFZOCST+0072.lfa.mh)
2022-11-22 00:00 + 3 J, 1 dom., 105 niv.



Tephigram, G6ZJ: potential temperature , lon= 1.38034 lat= 43.57835
Analysis Fri 6.10.2023 00h UTC + 16h, Valid Fri 6.10.2023 16h UTC (16:06 LST)

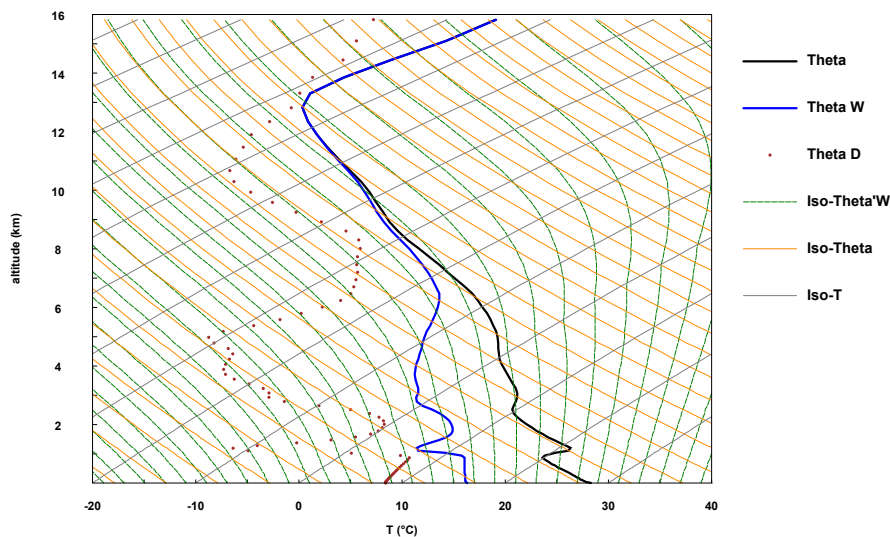


Table des matières

1	Main principles	5
1.1	User-defined horizontal domains	5
1.2	User-defined outputs	5
1.3	Scientific content of diagnostics	6
1.4	Output frequency	6
1.5	Internal representation of domains	6
1.6	Horizontal mean operator	6
2	Producing DDH files : geometry, user namelist	7
2.1	Different kinds of horizontal domains	7
2.1.1	Global domain	7
2.1.2	Zonal bands	7
2.1.3	Limited areas	8
2.2	Declarations. The NAMDDH namelist	10
2.2.1	Declarations	11
2.2.2	Declaration of limited domains	11
2.2.3	Default values	13
2.3	Output occurrence control	14
2.4	Identification of domains in the code and in the outputs	14
2.5	Logical structure of output files	15
2.5.1	File names	15
2.5.2	Articles giving information about dimensions and dates	15
2.5.3	Articles giving information about the type of domains	16
2.5.4	Articles giving information about scientific fields	17
3	Budget equations and horizontal mean	19
3.1	Generic budget equation	19
3.1.1	Term 1. Divergence of horizontal fluxes at the boundaries of the domain	19
3.1.2	Term 2. Divergence of the adiabatic vertical flux	20
3.1.3	Term 3. Adiabatic source term	20
3.1.4	Term 4. Physical fluxes divergence term	20
3.1.5	Term 5. Tendency term due to physical parametrizations	20
3.2	Horizontal mean	21
3.3	Temporal discretization	22
4	Budget and diagnostics, ARPEGE model	25
4.1	Moist air mass budget	25
4.2	Water mass budget	26
4.3	Momentum budget	27
4.4	Kinetic energy budget	28
4.5	Thermal energy budget	29
4.6	Angular momentum budget	30
4.7	Entropy budget	32
4.8	Potential energy budget	33
4.9	Surface budgets	34

4.10	Instantaneous diagnostics	35
4.10.1	Relative humidity	35
4.10.2	Cloudiness	35
4.10.3	Vertical velocity	35
4.10.4	Passive variables	36
4.10.5	Free style variables	36
4.10.6	Free style fluxes	36
4.11	Cumulated mass	36
5	Budgets and diagnostics, AROME model	37
5.1	Balance equations	37
5.1.1	Momentum	37
5.1.2	Turbulence kinetic energy	37
5.1.3	Thermal energy	38
5.1.4	Water vapour	38
5.1.5	Cloud water	39
5.1.6	Rain	39
5.1.7	Cloud ice	40
5.1.8	Snow	40
5.1.9	Graupel	41
5.2	Common Dynamics-Physics Interface CDPI	41
6	Using DDH files : the ddhtoolbox	43
6.1	Purpose	43
6.2	Install the software	43
6.3	Environment variables	43
6.4	Synoptic view of ddhtoolbox utilities	44
6.4.1	ddhr : get autodocumentation	45
6.4.2	ddht : transform	45
6.4.3	ddhi : interpretation	45
6.4.4	ddhb : budgets of prognostic variables	46
6.5	Using the ddhtoolbox, step by step	47
6.5.1	Vertical profile of variables, tendencies or fluxes	47
6.5.2	Plotting tephigrams from DDH files	50
6.5.3	Fluxes, with / without conversion to tendencies	52
6.5.4	Plotting ARPEGE/AROME budgets : example of studying infra-red / turbulence interaction	55
6.5.5	Plotting ARPEGE/AROME budgets : customizing the budget names and categories	57
6.5.6	Hovmöller diagrams of the difference between an experiment and a reference	57
7	Software maintenance	61
7.1	Main arrays and their organization	61
7.1.1	Arrays describing domains	61
7.1.2	Data arrays	61
7.1.3	Main local arrays	63
7.2	Organization of the main functions	64
7.3	How to add new fields to budgets	66
7.4	New dataflow for DDH	66
7.4.1	Introduction	66
7.4.2	Achievements-Future developments	66
7.4.3	General basics of the new dataflow	66
7.4.4	Activating and modifying the new dataflow	68
7.4.5	Architecture of the code	69
7.4.6	Remaining issues specific to the new dataflow	70
7.4.7	Conclusion	71
8	History	73

Chapitre 1

Main principles

This paper presents the diagnostics on horizontal domains (DDH) developed initially for the variable mesh of ARPEGE(global stretched model), and now also available for ALADIN (Limited Area Model) and AROME (Cloud Systems Resolving Model).

The main objective of the DDH tool is to provide, on user defined domains, the budget of prognostic variables of these models (momentum, temperature, water vapour, etc). The DDH tool is used by searchers and model developpers to understand the model's dynamical and physical interactions, thus contributing to the parameterization development process. The DDH are also used for other purposes, like getting mean model drifts with respect to analyses, or to extract model vertical profiles at given locations.

These diagnostics are made, on the one side, of logical functions which enable to manage several user defined horizontal domains depicting parts or the whole of the domain of integration, while reducing the number of scientific computations. On the other hand, they include the production of budget equations of the prognostic variables for domains such as zonal bands, rectangular areas, single vertical model columns, or the whole globe.

Each model point is described within the DDH software by its geographical position, a scale factor, the orientation of the geographical North vector, a mean value of each variable as well as some horizontal derivatives.

Each point is independent and can belong to different user-defined domains ; any mean on several points makes a domain. The gathering of all points makes the global domain. A zonal mean is made on a grouping of points between two given latitudes, etc.

DDH therefore makes a double representation of domains : a first one, external, which operates groupings meaningful for users ; that is a zonal band, or points within a given geographical area. The other one, internal, consists in the grouping of any other points, not even necessarily connex ones.

Information within a given domain is also categorized by DDH : for some parameters, an instantaneous value is required, for others, only a mean value in time is required. This implies the use of operators of horizontal means and of simple and linear mean values in time, commutative as often as possible.

1.1 User-defined horizontal domains

- whole globe,
- zonal bands of equal surfaces,
- limited domains, defined by either two or four corners, overlapping or not,
- isolated points, either inside or outside the above mentioned.

In practice, for the DDH software, limited domains and isolated points are the same kind.

1.2 User-defined outputs

- a print on standard output of budgets on any given domain, in vertical mean.
- the production of a file such as LFA which includes vertical profiles of mean horizontal parameters, eventually cumulated in time, on this ensemble of domains. Thus, the following files can be produced :
 - a file for the whole model domain,

- a file containing zonal bands,
- a file of limited domains and of isolated points,

1.3 Scientific content of diagnostics

- Budgets of mass / energy : balance of air and water masses, potential and internal energies, kinetic energy, momentum budget, entropy budget, can be activated from independent logical indicators.
- For surface, the DDH tool provides surface fluxes, but not soil budgets, as these soil budgets are given by the SURFEX software itself.

1.4 Output frequency

For DDH diagnostics, two independent control chains are available. One for printing, the other one for producing files. They enable to choose a time unit (time step or prediction time) and to generate output either at regular intervals or at specific irregular prediction times.

1.5 Internal representation of domains

The main principle of DDH is to allocate each model point to one and only one internal domain, and this even if this point belongs to several *user* domains. With this principle, scientific computations are made only once, thus saving computation time. Therefore, two ways to split the integration domain are used :

- the external (or user-defined) splitting : a given point can be at the same time, within the global domain, within a zonal band, and within one or more limited domains. Of course, final file or listing outputs are presented in this user-defined geometry, but one should try to make the least possible computations.
- the internal splitting, meaning the partition (in the sense of the ensemble theory) of the integration domain. Therefore, to each point is associated a single internal domain. The associated mask is set in the same way as the geographical positions' one, land/sea indicator, etc. To each user's domain corresponds a single set of internal domains. This set entirely defines the external domain.

1.6 Horizontal mean operator

It must be such that the average of the mean values of each sub ensemble gives the global mean value on the sphere (or on the entire model domain, in the case of a limited area model). Each point is given a weight which represents its geographical surface (but without dimension).

Chapitre 2

Producing DDH files : geometry, user namelist

This chapter presents two aspects of the know-how : first of all what has to be declared in the user namelist to generate budgets and DDH files, and secondly the file structure.

2.1 Different kinds of horizontal domains

To each kind of horizontal domain (global, zonal band, limited area) is associated a logical indicator of activation, two output indicators (files and/or listing), and, for limited area domains, a geometrical identification longitude/latitude of the corners).

Recognized types :

- The global domain is activated if `LHDGLB` is true,
- Zonal bands are activated if `LHDZON` is true,
- Limited area domains or isolated points are activated if `LHDDOP` is true.

2.1.1 Global domain

- `LHDPRG`, true for printing.
- `LHDEFG`, true for producing a file.

2.1.2 Zonal bands

The total number of zonal bands is `NDHKD`. In case of file output, a single file will contain all the bands. In case of print, only one band will be printed ; this band should be specified (`NDHZRP`).

To summarize

- `LHDPRZ` true induces the printing of the latitude band number `NDHZPR`,
- `LHDEFZ` true induces the writing of a file,
- `NDHKD` specifies the number of latitude bands.

The principle of it is to divide the real sphere into `NDHKD` bands of geographical latitudes of equal surfaces. Each band is identified by its index `jkd` ($1 \leq jkd \leq NDHKD$).

Let the zonal band between geographical latitudes $\theta_{g,jkd}$ and $\theta_{g,jkd+1}$, and notating

$$\mu_{jkd} = \sin \theta_{g,jkd},$$

the equality of the surfaces leads to create the following suite of the Northern boundary of the bands :

$$NDHKD \times 2\pi a^2 (\mu_{jkd} - \mu_{jkd+1}) = 4\pi a^2,$$

with $\mu_1 = 1$. Therefore

$$\mu_{jkd} = 1 - (jkd - 1) \frac{2}{\text{NDHKD}}$$

The latitude of the zonal band jkd is

$$\bar{\mu}_{jkd} = 1 - \left(jkd - \frac{1}{2} \right) \frac{2}{\text{NDHKD}}$$

2.1.3 Limited areas

Several type of limited domains may be defined

- Type 1 : an isolated point defined by its indexes (jlon, jgl),
- Type 2 : a domain defined by its four corners identified by (geographical longitude, geographical latitude),
- Type 3 : a domain defined by two opposite corners, identified by (geographical longitude, geographical latitude),
- Type 4 : an isolated point defined by its geographical position.

Please note that, for the moment and for simplicity sake, affectation computations are based on computation of straight lines in space (λ, μ) .

The management of either total or partial overlapping between domains calls for the notion of virtual plane. As a matter of fact, to each declared domain is associated a virtual plane. Thus, diagnostics DDH will know either

- that possible overlapping should be ignored and that diagnostics should refer to domains as they have been declared. To do so, domains must be located inside separate virtual planes,
- Or that overlapping is a way to modify the geometry of a domain already declared, therefore, diagnostics will refer to modified domains. To do so, domains must be affected to the same virtual plane, furthermore, the order of declaration is important. This will enable to create, in principle, any shapes of domains, from those elementary types.

Let us see a few examples.

Case 1.

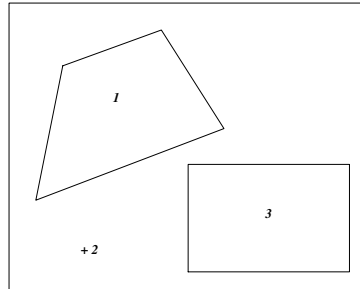


FIGURE 2.1 – Case 1

All domains are disjoint (figure 2.1). The notion of virtual plane is needless. In practice, put all the domains in the same plane.

Case 2.

Domain 4 overlaps domains 1, 2 and even 3 (figure 2.2). Complete results in each of these domains are required. In practice, allocate domains to distinct planes (figure 2.3).

Case 3.

Domains are embedded to up to three levels and complete results in each of them are required (figure 2.4). Allocate domains to distinct planes.

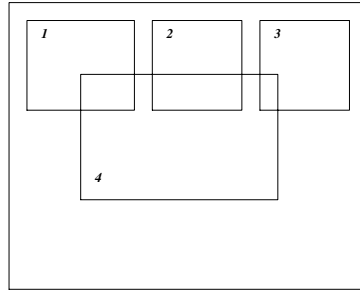


FIGURE 2.2 – Case 2

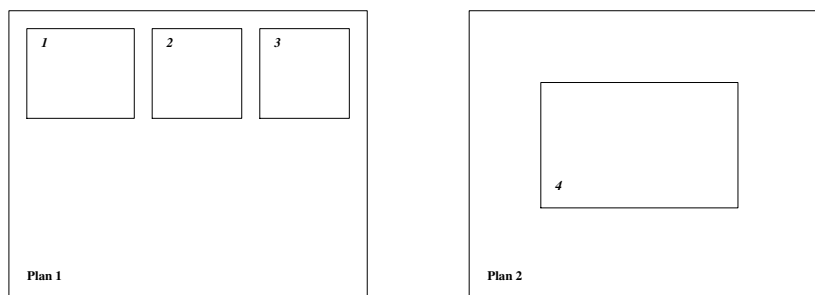


FIGURE 2.3 – Case 2

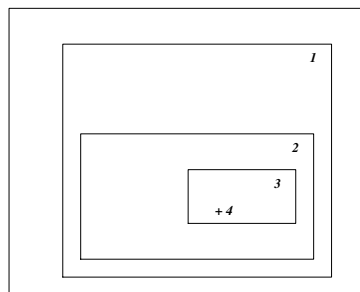


FIGURE 2.4 – Case 3

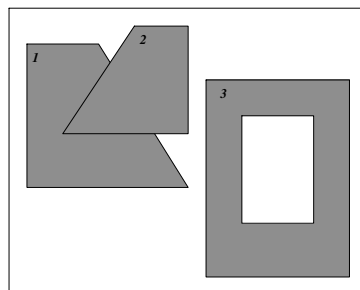


FIGURE 2.5 – Case 4

Case 4.

A domain is not a quadrilateral, another has the shape of a ring (figure 2.5). To get diagnostics in these domains, do the declarations in the same virtual plane, according to the numbers of figure 2.6.

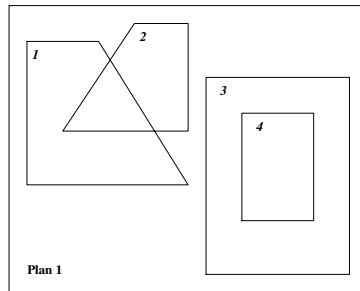


FIGURE 2.6 – Case 4

Some remarks about declaring domains. First of all, there is no algorithmic limit to the number of overlapping levels. For practical reasons, the limit is given by `PARAMETER JPDXHPU`, which is (quite) easy to modify.

There is no need to execute a formal declaration of a virtual plane (hence its name). In practice and from the point of view of users, it is only a coordinate of a domain. Allocating a domain to the plane N creates the plane N . However, memory and time costs can be lessened if affectations are declared to successive planes (1, 2, 3, ... and not 1, 3, 5, ... which would leave planes 2 and 4 empty).

Case 4 shows how declarations are managed within a same plane. To each virtual plane is associated a mask of points distribution. The index of a point is the number of the domain in the plane, 0 if the point does not belong to any domain. The declaration of a domain of the kind 1 to 4 being read and verified, points will be allocated to it, independently of their indexes at that time, 0 or any other value.

Thus, let us go back to case 4. Quadrilateral 1 is declared as a guess for domain 1. Every single points included in this quadrilateral is given index 1. Then, quadrilateral 2 is declared and its points are given index 2, notably, those which previously had value 1. The shape of points « 1 » changes and takes the shape of the final domain 1. Conversely, to get ring 3, it is sufficient to declare its outer outline and to make a hole inside it and then to declare its inner outline. *Within any given virtual plane, a point is allocated solely to the last declared domain which contains it.*

To summarize :

Embedded or overlapping domains : allocate each domain to a distinct virtual plane. Require as many virtual planes as it is necessary.

Strange or punched domains : within the same virtual plane, distort and make holes in the successive sketches of the quadrilateral.

The management system can not guess by itself which of the two approaches you will use. Therefore, should you not pay any attention to the declarations, especially to the virtual coordinate, an error may occur.

Besides, let the absent-minded reader be reminded that, even if a point seems to be relevant for the four domains, diagnostics will be made only once. Virtual planes are given up by the software in favour of a unique distribution plane where each new possible intersection makes a new "internal" computation domain. Thus, case number 1 gives (without the horizontal means) four internal domains, case number 2 (always alone) gives seven internal domains, and so forth.

Let us now present what has to be declared in order to start the DDH diagnostics.

2.2 Declarations. The `NAMDDH` namelist

As always with ARPEGE, setting up options is done by a namelist. For these diagnostics, most items depend on `NAMDDH`. However, the control of output events depends on `NAMCT0` and `NAMCT1`. Some dimensions, presently coded as `PARAMETER` could be managed more flexibly through `NAMDIM`.

2.2.1 Declarations

NAMDDH regroupes the main parameters controlling the diagnostics. Can be found logical indicators, some numerical parameters and also a table to declare possible limited domains.

Type of domains

- LHDGLB: global domain (diagnostics are produced if the indicator is true)
- LHDZON: zonal bands
- LHDDOP: limited domains and isolated points.

Variables to budgetise

- LHDHKS: budget of atmospheric mass, energy, momentum, relative humidity, soil budget.
- LHDMCI: budget of kinetic momentum
- LHDENT: budget of entropy

Should no domain be specified, no diagnostics are produced. Should a domain be specified, but no content specified, ARPEGE **is stopped**. The same goes if no output is requested.

Output on file or listing

- LHDEFG: write global diagnostics on file,
- LHDEFZ: write zonal bands diagnostics on file,
- LHDEFD: write limited domains diagnostics on file,
- LHDPRG: write global diagnostics on listing,
- LHDPZ: write zonal bands (a single band will be written) diagnostics on listing,
- NDHZPR: index of the latitude band whose budget will be printed (if LHDPZ is true),
- LHDPD: write limited domains diagnostics on listing,
- LHDFIL: the list of articles written in each DDH output file, will be written on listing.

Software maintenance, debugging mode

- LHDLIST: printing on listing in verbose mode,
- LHDVRF: verification mode, activating the budget computation in one point; the output is written on listing.
- NVDHLO: Index JLON of the verification point,
- NVDHGL: Index JGL of the verification point.

Control results reproducibility

- LHDREP: true if one wishes the results of the diagnostics to be reproducible bit to bit from a multitask run to the next one. This option, useful for data processing validation, is useless for scientific interpretation. Difference in « non reproducible » mode (LHDREP false) are not significant: to this day, no difference whatsoever, up to 10^{-10} in the relative way, has been noticed! The advantage of the recommended option LHDREP = .FALSE. is to make substantial savings in the occupation of the memory.

Number of zonal bands

- NDHKD: Number of zonal bands.

2.2.2 Declaration of limited domains

The declaration is made by filling in a double entry table BDEDDH (10, JPDHNOX). JPDHNOX is a PARAMETER which gives the maximum number of possible limited domains. It goes together with JPDHXPU, maximum number of virtual planes.

For each domain, its type, its virtual plane and indications dependant on the type are given. Therefore

$$\text{BDEDDH}(1, \text{domain number}) = \text{type}$$

$$\text{BDEDDH}(2, \text{domain number}) = \text{virtual plane}$$

For type 1, point given by its indexes

$$\text{BDEDDH}(3, \text{domain number}) = \text{rjlon}$$

$$\text{BDEDDH}(4, \text{domain number}) = \text{rjgl}$$

For type 2, quadrilateral given by its four corners

$$\text{BDEDDH}(3, \text{domain number}) = \text{Longitude of corner \#1, in degrees, } \lambda_1$$

$$\text{BDEDDH}(4, \text{domain number}) = \text{Latitude of corner \#1, in degrees, } \theta_1$$

$$(\text{BDEDDH}(5, -), \text{BDEDDH}(6, -)) = (\lambda_2, \theta_2)$$

$$(\text{BDEDDH}(7, -), \text{BDEDDH}(8, -)) = (\lambda_3, \theta_3)$$

$$(\text{BDEDDH}(9, -), \text{BDEDDH}(10, -)) = (\lambda_4, \theta_4)$$

In order to specify a domain, one must comply with the following constraints :

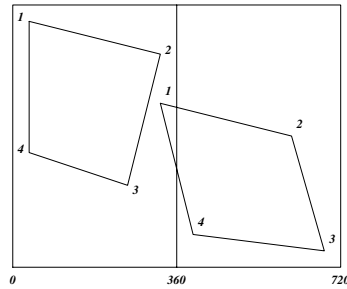


FIGURE 2.7 – Constraints for quadrilateral declarations

— For a domain which does not intersect the Greenwich meridian

$$-1 \leq \mu_i = \sin \theta_i \leq 1, \quad 0 \leq \lambda_i \leq 360^\circ$$

and

$$\lambda_2 > \lambda_1, \quad \mu_3 < \mu_2, \quad \lambda_3 > \lambda_4, \quad \mu_1 > \mu_4$$

— For a domain which intersects the Greenwich meridian, the same order in the declaration of corners applies, but furthermore, we have

$$(\lambda_1 \leq 360^\circ \text{ ou } \lambda_4 \leq 360^\circ) \quad \text{and} \quad (\lambda_2 > 360^\circ \text{ ou } \lambda_3 > 360^\circ)$$

For type 3, rectangular domain given by two opposite corners

$$\text{BDEDDH}(3, \text{domain number}) = \text{Longitude of corner \#1, in degrees, } \lambda_1$$

$$\text{BDEDDH}(4, \text{domain number}) = \text{Latitude of corner \#1, in degrees, } \theta_1$$

$$(\text{BDEDDH}(5, -), \text{BDEDDH}(6, -)) = (\lambda_3, \theta_3)$$

This means that, with the same constraints as above, one declares only corners 1 and 3. Implicitly

$$(\lambda_2 = \lambda_3, \mu_2 = \mu_1) \quad \text{et} \quad (\lambda_4 = \lambda_1, \mu_4 = \mu_3)$$

For type 4, points given by their geographical position

$$\text{BDEDDH}(3, \text{domain number}) = \text{longitude } \lambda_g \text{ in degrees}$$

$$\text{BDEDDH}(4, \text{domain number}) = \text{latitude } \theta_g \text{ in degrees}$$

In that case, diagnostics will be made on the closest grid point, using the spatial Euclidian metric in the (λ, μ) space.

Allocation of points to the domains

The point whose geographical coordinate is (λ_g, μ_g) is allocated to a type 2 or 3 domain in the following conditions.

Domain which does not intersect the Greenwich meridian ($\lambda_2 \leq 2\pi$ et $\lambda_3 \leq 2\pi$)

$$\mu_g \leq \frac{\mu_2 - \mu_1}{\lambda_2 - \lambda_1} \lambda_g + \frac{\mu_1 \lambda_2 - \mu_2 \lambda_1}{\lambda_2 - \lambda_1}$$

$$\lambda_g \leq \frac{\lambda_3 - \lambda_2}{\mu_3 - \mu_2} \mu_g + \frac{\mu_3 \lambda_2 - \mu_2 \lambda_3}{\mu_3 - \mu_2}$$

$$\mu_g \geq \frac{\mu_4 - \mu_3}{\lambda_4 - \lambda_3} \lambda_g + \frac{\mu_3 \lambda_4 - \mu_4 \lambda_3}{\lambda_4 - \lambda_3}$$

$$\lambda_g \geq \frac{\lambda_1 - \lambda_4}{\mu_1 - \mu_4} \mu_g + \frac{\mu_1 \lambda_4 - \mu_4 \lambda_1}{\mu_1 - \mu_4}$$

Domain overlapping $\lambda = 0$ The same tests must be made on $(\lambda_g + 2\pi, \mu_g)$ for any point so that $\lambda_g \leq \max(\lambda_3 - 2\pi, \lambda_2 - 2\pi)$.

2.2.3 Default values

By default, all logical indicators are FALSE and dimensions are set to zero. By default, no DDH is done.

Here is an example of a namelist which activates diagnostics in all types of domains. 6 zonal bands are requested. 6 limited domains are declared in 3 virtual planes. Domain 2 is intersecting the Greenwich meridian.

```

NAMDDH
LHDGLB = .TRUE., LHDZON = .TRUE., LHDDOP = .TRUE., LHDHKS = .TRUE.,
LHDMCI = .FALSE., LHDENT = .FALSE., LHDPRG = .TRUE., LHDPRD = .TRUE.,
LHDPRZ = . TRUE. NDHZPR = 3,
LHDEFG = .TRUE., LHDEFZ = .TRUE., LHDEFD = .TRUE., LHDLIST = .TRUE.,
NDHKD = 6, LHDVRF = .TRUE., NVDHLO = 5, NVDHGL = 29,
BDEDDH(1,1) = 2., BDEDDH(2,1) = 1., BDEDDH(3,1) = 250., BDEDDH(4,1) = 45.,
BDEDDH(5,1) = 440., BDEDDH(6,1) = 80., BDEDDH(7,1) = 360., BDEDDH(8,1) = 15.,
BDEDDH(9,1) = 215., BDEDDH(10,1) = 35.,
BDEDDH(1,2) = 2., BDEDDH(2,2) = 2., BDEDDH(3,2) = 125., BDEDDH(4,2) = 20.,
BDEDDH(5,2) = 350., BDEDDH(6,2) = 75., BDEDDH(7,2) = 360., BDEDDH(8,2) = -5.,
BDEDDH(9,2) = 85., BDEDDH(10,2) = -20.,
BDEDDH(1,3) = 3., BDEDDH(2,3) = 1., BDEDDH(3,3) = 30., BDEDDH(4,3) = 10.,
BDEDDH(5,3) = 245., BDEDDH(6,3) = -30.,
BDEDDH(1,4) = 3., BDEDDH(2,4) = 2., BDEDDH(3,4) = 0., BDEDDH(4,4) = 90.,
BDEDDH(5,4) = 359., BDEDDH(6,4) = 85.,
BDEDDH(1,5) = 1., BDEDDH(2,5) = 1., BDEDDH(3,5) = 10., BDEDDH(4,5) = 1.,
BDEDDH(1,6) = 4., BDEDDH(2,6) = 2., BDEDDH(3,6) = 20., BDEDDH(4,6) = -70.,

```

2.3 Output occurrence control

As for other ARPEGE output, regular output frequency may be given in time step. One may also fill in a table which will enable to make irregular outputs.

The control tables are N_{xTS} (0 : JPNPST) with $x=DHFG$ for file outputs of the global domain, $x=DHFZ$ for file outputs of zonal band domains, $x=DHFD$ for file outputs of limited area domains, and $x=DHP$ for printed outputs.

Time units are NFR_x .

Tables and units are initialized through $NAMCT0$.

Three kinds of outputs are possible

1. If $NFR_x = n$ ($n > 0$) and $N_{xTS}(0) = 0$: output every n time steps.
2. If $NFR_x = n$ ($n > 0$), $N_{xTS}(0) = m$ ($m > 0$) and $N_{xTS}(i) = p_i$ ($p_i \geq 0, i \in \{1, \dots, m\}$) : output at time steps np_i .
In this case it will be preferable to set $LINC = .FALSE.$ in namelist $NAMOPH$, in order to force the date units in output file names to be in time steps (rather than in hours).
3. If $NFR_x = n$ ($n > 0$), $N_{xTS}(0) = -m$ ($m > 0$) and $N_{xTS}(i) = -p_i$ ($p_i \geq 0, i \in \{1, \dots, m\}$) : output at hours np_i .

As a matter of fact, these outputs are only possible at these time steps. They are actually produced if, in addition, $N1_x = 1$.

These parameters belong to $MODULE/YOMCT1/$, initialized in $SU1YOM$ with the namelist $NAMCT1$. They are set to zero, if either diagnostics DDH are not activated or if no file or no listing are requested. This cancels the corresponding output, whatever the content of $NAMCT0$ may be.

2.4 Identification of domains in the code and in the outputs

The user-type identification of domains ($BDEDDH$) is transformed in a simpler form, for use by the internal part of the DDH software. Here are indicated the identification conventions which are used internally by the DDH . To each domain is associated a descriptor of 11 words.

- Words 1 and 2 are the coordinates of the domain : virtual plane and number in the plane,
- Words 3 to 10 are mostly geographical information type dependant,
- Word 11 is the kind of domain.

DOMAIN IDENTIFIER										
1	2	3	4	5	6	7	8	9	10	11
Virtual plane	number in this plane	Type-dependent information								Type
plan	number	λ_g	μ_g	r_{jlon}	r_{jgl}					Type 1. Point
plan	number	λ_g	μ_g	r_{jlon}	r_{jgl}	λ_g user value	μ_g user value			Type 4. Point
plan	number	λ_1	μ_1	λ_2	μ_2	λ_3	μ_3	λ_4	μ_4	Type 2. Quadrilateral
plan	number	λ_1	μ_2	λ_2	$\mu_2=\mu_1$	$\lambda_3=\lambda_2$	μ_3	$\lambda_4=\lambda_1$	$\mu_4=\mu_3$	Type 3. Rectangle
0	0									Type 5. Globe
0	jkd	ndhkd	$\bar{\mu}_{jkd}$							Type 6. Zonal band

Non allocated values are initialized to zero. Longitudes are in radian. For points and limited domains, this information is kept in table FNODDH (11, JPDHNOX) of the MODULE /YOM1DDH/. For the globe and zonal bands, the information is really useful only during the output.

Moreover, the properties of the domains are the ones declared by the user (except for 4) : a way to show that a domain has then been deformed or punched is still lacking.

2.5 Logical structure of output files

For a given date, files contain a suite of domains. For each of them, a suite of profiles and soil variables can be found. Three files can be produced : global, zonal, limited area. These files are physically written with the LFA software (Jean-Marcel Piriou), if LHDLFA is true, and in pseudo-GRIB format, if LHDLFA is false.

2.5.1 File names

Global DHFGLeeee+nnnn

Zonal bands DHFZOeeee+nnnn

Limited area domains DHFDLeeee+nnn

eeee : the first 4 letters of the name of the experiment,

nnnn : output date in hour or time step, according to the logical indicator LINC from namelist NAMOPH.

2.5.2 Articles giving information about dimensions and dates

Article 1.

The first physical article 'INDICE EXPERIENCE' contains the name of the experiment.

Article 2.

Article 'DATE' (11 mots).

- 1 . Year,
- 2 . Month,
- 3 . Day,

- 4 . Hour,
- 5 . Minute, date of integration start.
- 6 . 1 if forecast range is in hours, 2 if forecast range is in days,
- 7 . Forecast range,
- 8 . 0,
- 9 . 10, except maybe at the beginning,
- 10 . Number of cumulated values,
- 11 . 0.

Article 3.

Article 'DOCFICHIER' (17 words).

- 1. File type :
 - 1 limited area domains,
 - 5 global domain,
 - 6 zonal bands.
- 2. 0 if LHDHKS is false, 1 if true,
- 3. 0 if LHDMCI is false, 1 if true,
- 4. 0 if LHDENT is false, 1 if true,
- 5. NSTEP, current time step value,
- 6. NFLEV, number of levels. Length of variable profiles or variable tendencies. The length of the flux profiles is NFLEV+1,
- 7. NDHCV, total number of vertical profiles for each domain,
- 8. NDHCS, total number of soil fields,
- 9. NDHVV, number of variable profiles at a given time. The file contains 2 instantaneous variables : the initial one and that of current time step.
- 10. NDHFVD, number of « dynamical » fluxes or tendencies in vertical profiles,
- 11. NDHFVP, number of « physical » fluxes or tendencies in vertical profiles,
- 12. NDHVS, number of instantaneous soil variables,
- 13. NDHFSD, number of soil « dynamical » fluxes,
- 14. NDHFSP, number of soil « physical » fluxes,
- 15. number of domains in the file :
 - 1 for the globe,
 - NDHKD for zonal bands,
 - NDHNOM for limited area domains.
- 16. number of "free" soil variables : these variables are used at ECMWF for diagnostics such as 10 *m* winds, roughness, etc.
- 17. number of "free" soil fluxes.

Article 4.

Article 'ECHEANCE' forecast range in seconds (1 word).

2.5.3 Articles giving information about the type of domains

For each domain, there is an identification article 'DOCDnnnn', where nnn is the name of the domain. This article is made of 11 words whose content has been described page 14.

2.5.4 Articles giving information about scientific fields

The last part of this documentation will be about the definition of each field in each option as well as the name of this field. Here, we only will indicate how the name of articles are constituted.

The name of articles takes the form `nnntvvssssssssss`, with

`nnn` : number of the domain in the file. `nnn` varies from 1 to `DOCFICHER(17)`.

`t` : type of field contained in the article :

`V` : variable profile, length `NFLEV`,
`T` : tendency profile, length `NFLEV`,
`F` : flux profile, length `NFLEV+1`,
`S` : soil data, length : cf. page 34.

`vv` : physical variable written in this file article :

`PP` : pressure,
`QV` : specific water vapour content ,
`UU` : zonal momentum,
`VV` : merional momentum,
`KK` : kinetic energy,
`CT` : thermal energy,
`EN` : entropy,
`M1` : angular momentum,
`EP` : potential energy ($\Phi = g z$).

The next 10 characters (suffix) make the field specific name. However, some general rules do also apply : for variables given as profiles (whose name is therefore `VV`, the instant must be indicated

`ssssssssss = 0` variable at initial time step,
`ssssssssss = 1` variable at current time step.

Some suffixes crop up quite frequently

`ssssssssss = DIVFLUHOR` for terms of the kind $\text{div}_\eta \left(\chi \vec{v} \frac{\partial p}{\partial \eta} \right)$
`ssssssssss = FLUVERTDYN` for terms $\chi \dot{\eta} \frac{\partial p}{\partial \eta}$,
`ssssssssss = FLUDUAPLUI` for terms $\delta_m F_p \chi$.

Chapitre 3

Budget equations and horizontal mean

This chapter is about budget equations and discretization, in space and time.

3.1 Generic budget equation

Let χ be a variable of the model whose budget is required. The generic form of the χ budget may be written

$$\frac{\partial}{\partial t} \left(\chi \frac{\partial p}{\partial \eta} \right) = \underbrace{-\operatorname{div}_\eta \left(\chi \vec{v} \frac{\partial p}{\partial \eta} \right)}_{(1)} - \underbrace{\frac{\partial}{\partial \eta} \left(\chi \dot{\eta} \frac{\partial p}{\partial \eta} \right)}_{(2)} + \underbrace{S_d \frac{\partial p}{\partial \eta}}_{(3)} - \underbrace{g \frac{\partial F_\varphi}{\partial \eta}}_{(4)} - \underbrace{g S_\varphi \frac{\partial G_\varphi}{\partial \eta}}_{(5)}$$

To estimate the budgets, it has been decided to systematically work on the extensive scales $\chi \frac{\partial p}{\partial \eta}$, that is to say, $\chi_\ell \delta p_\ell$ for the discrete value in the layer ℓ . In practice, the application of the vertical discretization leads this equation to

$$\frac{\partial}{\partial t} (\chi \delta p) = -\operatorname{div}_\eta (\chi \vec{v} \delta p) - \delta \left(\chi \dot{\eta} \frac{\partial p}{\partial \eta} \right) + S_d \delta p - g \delta F_\varphi - g S_\varphi \delta G_\varphi$$

where every term is indexed by ℓ , index of the layer of the model for which this equation means something/makes sense/. The operator $\delta \xi_\ell$ is

$$\delta \xi_\ell = \xi_{\tilde{\ell}} - \xi_{\tilde{\ell}-1}$$

where $\xi_{\tilde{\ell}}$ takes the value of ξ at the interlayer $\tilde{\ell}$.

3.1.1 Term 1. Divergence of horizontal fluxes at the boundaries of the domain

In order to be computed, this term needs to know the χ gradient. For the initial conditions this will not always be the case. This term will not be complete, especially every time when χ depends on the momentum (momentum itself, kinetic energy, angular momentum, etc). Whenever possible,

$$-\chi \left(\delta p \operatorname{div} \vec{v} + \delta B \vec{v} \vec{\nabla} \pi \right) - \delta p \vec{v} \vec{\nabla} \chi$$

is computed. The first term can always be computed. Term 1 must be null in global mean. For a band of latitude, it gives the value of the divergence of the meridian flux χ .

3.1.2 Term 2. Divergence of the adiabatic vertical flux

This term will be treated as a flux : the horizontal mean of the quantity will be kept

$$\left(\chi \dot{\eta} \frac{\partial p}{\partial \eta} \right)_{\tilde{\ell}}$$

As in the discretization of vertical advection terms,

$$\left(\chi \dot{\eta} \frac{\partial p}{\partial \eta} \right)_{\tilde{\ell}} = \frac{1}{2} (\chi_{\ell} + \chi_{\ell+1}) \left(\dot{\eta} \frac{\partial p}{\partial \eta} \right)_{\tilde{\ell}} \quad \text{pour } \tilde{\ell} = 0, \dots, NFLEV$$

will be computed.

The vertical speed $\dot{\eta} \partial p / \partial \eta$ is computed by GPCTY and modified by the lower boundary conditions.

3.1.3 Term 3. Adiabatic source term

Some terms of this kind can be deduced from the dynamical code. For example, the potential and the internal budget express the term called conversion term

$$\left[S_d \frac{\partial p}{\partial \eta} \right]_{c_p T} = -\frac{1}{g} \vec{v} \left[\vec{\nabla} \Phi + \frac{RT}{p} \vec{\nabla} p \right] \frac{\partial p}{\partial \eta}$$

Some similar terms are to be found in the kinetic energy budget. They, as term 1, are in the « tendencies » category, expressed at ℓ levels.

3.1.4 Term 4. Physical fluxes divergence term

Physical fluxes $F_{\varphi_{\ell}}$ are horizontally averaged as such. The thermal energy flux due to precipitations is the only tricky one. Formally, the following form is assumed

$$F_{\varphi_{c_p T}} = L(\eta, T) F_{\varphi_q}^{precip}(\eta)$$

Some assumptions need to be introduced, like

$$L(T_{\ell}) = L \left(\frac{1}{2} (T_{\ell} + T_{\ell+1}) \right)$$

where L is an "effective" latent heat, or difference in enthalpy due to phase change.

3.1.5 Terme 5. Tendency term due to physical parametrizations

Such terms occur in the energy budgets, e.g. the dissipation term

$$\vec{v} \frac{\partial F_{\varphi_v}^{tur+conv}}{\partial \eta}$$

or in the entropy budget

$$\frac{1}{T} \frac{\partial F_{\varphi_T}^{ray}}{\partial \eta}$$

These terms are computed, using variables at time t . Budget terms gather into three categories :

- variables, from the $\frac{\partial}{\partial t} \chi \delta p$ term,
- tendencies, at model levels (such as $\text{div}(\chi \delta p \vec{v})$),
- fluxes, at the inter-layers (δF_{χ}).

One shows below the discretization process, on the simplified budget equation

$$\frac{\partial}{\partial t} \left(\frac{1}{g} \chi \delta p \right) = \left(\frac{1}{g} \chi \delta p \right) \text{tend} - \delta F_\chi$$

which shows three categories. The goal of DDH diagnostics is to give information on the mean budget on an horizontal domain \mathcal{D} (surface $S_{\mathcal{D}}$) :

$$\frac{1}{S_{\mathcal{D}}} \frac{\partial}{\partial t} \iint_{\mathcal{D}} \frac{1}{g} \chi \delta p d\sigma = \frac{1}{S_{\mathcal{D}}} \iint_{\mathcal{D}} \left(\frac{1}{g} \chi \delta p \right) \text{tend} d\sigma - \frac{1}{S_{\mathcal{D}}} \iint_{\mathcal{D}} \delta F_\chi d\sigma$$

Some terms, such as the effect of the horizontal diffusion, cannot be diagnosed by DDH : horizontal diffusion is computed in spectral mode, is not converted into grid-point space, and thus unavailable for DDH.

3.2 Horizontal mean

Let $[\chi]_G$ be the global mean. We have

$$[\chi]_G = \frac{1}{4\pi a^2} \int_0^{2\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \chi a^2 \cos \theta_g d\theta_g d\lambda_g = \frac{1}{4\pi} \int_0^{2\pi} \int_{-1}^1 \chi d\mu_g d\lambda_g$$

For a given truncation, a precise quadrature of this relation reads

$$[\chi]_G = \sum_{k=1}^K \frac{1}{J(k)} \sum_{j=1}^{J(k)} \varpi_k \chi_{j,k} \quad \text{with} \quad K \geq \frac{3N+1}{2}$$

where N is the triangular truncation, ϖ_k is Gauss weight and $J(k)$ the number of points on the circle of latitude k .

Thus, k refers to the latitudes of the gaussian grid and j to the longitudes. On the stretched sphere the scale factor is a function of the spherical harmonics :

$$[\chi]_G = \frac{1}{S_G} \sum_{k=1}^K \sum_{j=1}^{J(k)} \chi_{j,k} \frac{\varpi_k}{J(k)m_{j,k}^2}$$

now with

$$S_G = \sum_{k=1}^K \sum_{j=1}^{J(k)} \frac{\varpi_k}{J(k)m_{j,k}^2} \neq 1$$

Therefore, the weight of each point will be assimilated as, for the present case, a non dimensional « area element » $\sigma_{j,k}$

$$\sigma_{j,k} = \frac{\varpi_k}{J(k)m_{j,k}^2}$$

Surface of domain \mathcal{D} :

$$S_{\mathcal{D}} = \sum_{(j,k) \in \mathcal{D}} \sigma_{j,k}$$

The horizontal mean of parameter χ on \mathcal{D} is written

$$\left[\frac{1}{S_{\mathcal{D}}} \iint_{\mathcal{D}} \chi d\sigma \right] = [\chi]_{\mathcal{D}} = \frac{1}{S_{\mathcal{D}}} \sum_{(j,k) \in \mathcal{D}} \chi_{j,k} \sigma_{j,k}$$

With this definition, a division of the globe in D domains \mathcal{D} is such that

$$[\chi]_G = \frac{1}{S_G} \sum_{d=1}^D [\chi]_d S_d$$

The user domain \mathcal{D} is divided into one or several internal domains \mathcal{D}_i . For a multitask run on P processors, isolated P terms for every internal domain are computed, which means,

$$[\chi]_{\mathcal{D}_i} S_{\mathcal{D}_i} = \sum_{p=1}^P \left(\sum_{(j,k) \in \mathcal{D}_i(p)} \chi_{j,k} \sigma_{j,k} \right)$$

XX

where $\mathcal{D}(p)$ are the points belonging to \mathcal{D} treated in the p task.

If the domain is cut in P parts, identical in mono or multi tasks, the mean being a simple sum (products are made in each under task p), results become easily reproducible. To summarize, the output mean on the domain \mathcal{D} called for the user is

$$[\chi]_{\mathcal{D}} = \frac{1}{S_{\mathcal{D}}} \sum_{\substack{\mathcal{D}_i \\ \cup \mathcal{D}_i = \mathcal{D}}} \left[\sum_{p=1}^P \left(\sum_{(j,k) \in \mathcal{D}_i(p)} \chi_{j,k} \sigma_{j,k} \right) \right]$$

Where the two most external \sum signs can very easily switch over. The algorithm is thus the following :

- Parallel computations of the necessary quantities $\chi_{j,k}$ on every points, initial computation of $\sigma_{j,k}$ and transit through the physico-dynamical interface.
- « condensation » of results for a p task in parts of the internal domains it manages. Note that this non vectorisable operation is nevertheless field and level independent : it is therefore along this direction that vectorization will take place.
- Synthesis for output needs only (i. e. from time to time) of partial sums on user's domains.

3.3 Temporal discretization

The typical budget equation

$$\int_0^{nstep \times \delta t} \frac{\partial}{\partial t} \frac{1}{g} \chi \delta p dt = \int_0^{nstep \times \delta t} \left[\left(\frac{1}{g} \chi \delta p \right)_{\text{tend}} - \delta F_{\chi} \right] dt$$

is integrated by the diagnostics DDH as follow

$$\left(\frac{1}{g} \chi \delta p \right)^{nstep} - \left(\frac{1}{g} \chi \delta p \right)^0 = \delta t \sum_{jstep=0}^{nstep-1} \left[\left(\frac{1}{g} \chi \delta p \right)_{\text{tend}} - \delta F_{\chi} \right]^{jstep}$$

Where δt stands for `TSTEP`, the nominal time step, and `nstep` the number of the current time step. Tendencies and fluxes cumulated in time must be stopped at the tendencies and at the fluxes of the time step preceding the output moment `NSTEP`. Tendencies which are computed by `CPG` when the grid-point variable is known must not be added before the output of results. This adds an important constraint to the parallel treatment.

In practice, one uses two arrays : one with initial variables and variables cumulated in time up to `NSTEP-1`, and another with values at `NSTEP` and variables cumulated in time up to `NSTEP` (which leads to the variable d'état at `NSTEP+1`).

In short, the typical budget equation for a layer ℓ and a discretized domain \mathcal{D}

$$[\xi]_{\mathcal{D}} = \frac{1}{S_{\mathcal{D}}} \sum_{(j,k) \in \mathcal{D}} \xi_{(j,k)} \sigma_{(j,k)}$$

with

$$S_{\mathcal{D}} = \sum_{(j,k) \in \mathcal{D}} \sigma_{(j,k)} \quad \sigma_{(j,k)} = \frac{\varpi_k}{J(k) m_{j,k}^2}$$

$$\left[\frac{1}{g} \chi \delta p \right]_{\mathcal{D}}^{\ell} (t = NSTEP \times \delta t) - \left[\frac{1}{g} \chi \delta p \right]_{\mathcal{D}}^{\ell} (t = 0) = \delta t \sum_{n=0}^{nstep-1} \left[\left(\frac{1}{g} \chi \delta p \right) \text{tend} \right]_{\mathcal{D}}^{\ell} (n) +$$

$$\delta t \left(\sum_{n=0}^{nstep-1} [F_{\chi}]_{\mathcal{D}}^{\tilde{\ell}-1}(n) - \sum_{n=0}^{nstep-1} [F_{\chi}]_{\mathcal{D}}^{\tilde{\ell}}(n) \right)$$

and the vertical mean budget

$$\sum_{\ell=1}^{NFLEV} \left[\frac{1}{g} \chi \delta p \right]_{\mathcal{D}}^{\ell} (t = NSTEP \times \delta t) - \sum_{\ell=1}^{NFLEV} \left[\frac{1}{g} \chi \delta p \right]_{\mathcal{D}}^{\ell} (t = 0) =$$

$$\delta t \sum_{\ell=1}^{NFLEV} \sum_{n=0}^{nstep-1} \left[\left(\frac{1}{g} \chi \delta p \right) \text{tend} \right]_{\mathcal{D}}^{\ell} (n) + \delta t \left(\sum_{n=0}^{nstep-1} [F_{\chi}]_{\mathcal{D}}^0(n) - \sum_{n=0}^{nstep-1} [F_{\chi}]_{\mathcal{D}}^{NFLEV}(n) \right)$$

Chapitre 4

Budget and diagnostics, ARPEGE model

This chapter presents budget equations which are the first application of DDH. It gives information about the content of output files.

4.1 Moist air mass budget

Budget equation

$$\frac{\partial r_\eta}{\partial t} = -\text{div}_\eta (r_\eta \vec{v}) - \frac{\partial}{\partial \eta} (r_\eta \dot{\eta}) + \delta_m \frac{\partial F_p}{\partial \eta}$$

where

- $r_\eta = -\frac{1}{g} \frac{\partial p}{\partial \eta}$.
- $\left(\dot{\eta} \frac{\partial p}{\partial \eta} \right)_{\eta=0} = 0 \quad \left(\dot{\eta} \frac{\partial p}{\partial \eta} \right)_{\eta=1} = \delta_m g E$.
- $F_p = F_p^{conv-l} + F_p^{conv-n} + F_p^{stra-l} + F_p^{stra-n}$.
- $\delta_m = 0$: masse conserved, $\delta_m = 1$: variable mass.

File output

VPP0	$\frac{1}{g} \delta p (t=0)$	
VPP1	$\frac{1}{g} \delta p (t=\text{NSTEP } \delta t)$	
TPPDIVFLUHOR	$-\frac{\delta t}{g} \text{div}_\eta (\vec{v} \delta p)$	cumulated
FPPFLUVERTDYN	$\frac{\delta t}{g} \dot{\eta} \frac{\partial p}{\partial \eta}$	cumulated
FPPSUMFPL	$\delta_m \delta t F_p$	cumulated

If χ^* stands for a quantity modified by the mass exchange and if by χ the initial quantity, then

$$\left(\dot{\eta} \frac{\partial p}{\partial \eta} \right)_{\tilde{\ell}}^* = \left(\dot{\eta} \frac{\partial p}{\partial \eta} \right)_{\tilde{\ell}} + \delta_m g \left[B_{\tilde{\ell}} (F_{p\tilde{L}} + E) - F_{p\tilde{\ell}} \right]$$

One should have at the lower limit

$$F_{q\tilde{L}}^{tur} = E (1 - \delta_m q_v)$$

Furthermore

$$\left(\frac{\omega}{p}\right)_\ell^* = \left(\frac{\omega}{p}\right)_\ell - \delta_m g \frac{1}{\delta p_\ell} \left[\alpha_\ell \delta F_p + \ln \left(\frac{p_{\tilde{\ell}}}{p_{\tilde{\ell}-1}} \right) F_{p_{\tilde{\ell}-1}} \right]$$

$$\left(\frac{\partial \pi}{\partial t}\right)^* = \frac{\partial \pi}{\partial t} - \delta_m g (E + F_{p\tilde{L}})$$

4.2 Water mass budget

Lagrangian equation

$$r_\eta \frac{dq_\psi}{dt} = T_{q_\psi}^{diff-hor} + \frac{\partial F_{q_\psi}}{\partial \eta} - \delta_m q_\psi \frac{\partial F_p}{\partial \eta}$$

Budget equation

$$\frac{\partial(r_\eta q_\psi)}{\partial t} = -\text{div}_\eta(r_\eta q_\psi \vec{v}) - \frac{\partial}{\partial \eta}(r_\eta q_\psi \dot{\eta}) + \frac{\partial F_{q_\psi}}{\partial \eta}$$

where

- $q_\psi = q_v, q_l$ ou q_n .
- $r_\eta = -\frac{1}{g} \frac{\partial p}{\partial \eta}$.
- $F_{q_v} = F_c^{conv-l} + F_c^{conv-n} + F_c^{stra-l} + F_c^{stra-n} + F_{q_v}^{tur} + F_{q_v}^{tur-conv}$.
- $F_{q_l} = F_p^{conv-l} + F_p^{stra-l} - F_c^{conv-l} - F_c^{stra-l} + F_{q_l}^{tur} + F_{q_l}^{tur-conv}$.
- $F_{q_n} = F_p^{conv-n} + F_p^{stra-n} - F_c^{conv-n} - F_c^{stra-n} + F_{q_n}^{tur} + F_{q_n}^{tur-conv}$.
- $F_{q_\psi}^{tur}$ contains the correction $F_{q_\psi}^{q<0}$ of the negative water values created eventually by the dynamics.

The tendency term due to the horizontal diffusion is missing from the budget equation : this diffusion is done in spectral mode, the information is thus not accessible to the DDH grid-point diagnostics.

File output

FQTPRECICOL	$\delta t F_p^{conv-l}$	cumulated
FQTPRECICON	$\delta t F_p^{conv-n}$	cumulated
FQTPRECISTL	$\delta t F_p^{stra-l}$	cumulated
FQTPRECISTN	$\delta t F_p^{stra-n}$	cumulated
FQTCONDECOL	$\delta t F_c^{conv-l}$	cumulated if LHDQLN=.TRUE.
FQTCONDECON	$\delta t F_c^{conv-n}$	cumulated if LHDQLN=.TRUE.
FQTCONDESTL	$\delta t F_c^{stra-l}$	cumulated if LHDQLN=.TRUE.
FQTCONDESTN	$\delta t F_c^{stra-n}$	cumulated if LHDQLN=.TRUE.
VQV0	$\frac{1}{g} q_v \delta p$ (t=0)	
VQV1	$\frac{1}{g} q_v \delta p$ (t=NSTEP)	
TQVDIVFLUHOR	$-\frac{\delta t}{g} \text{div}_\eta(q_v \delta p \vec{v})$	cumulated
FQVFLUVERTDYN	$\frac{\delta t}{g} q_v \dot{\eta} \frac{\partial p}{\partial \eta}$	cumulated
FQVTUR	$\delta t F_{q_v}^{tur}$	cumulated
FQVTURCONV	$\delta t F_{q_v}^{tur-conv}$	cumulated
FQVTURQNEGAT	$\delta t F_{q_v}^{q<0}$	cumulated
VQL0	$\frac{1}{g} q_l \delta p$ (t=0)	if LHDQLN=.TRUE.
VQL1	$\frac{1}{g} q_l \delta p$ (t=NSTEP)	if LHDQLN=.TRUE.

TQLDIVFLUHOR	$-\frac{\delta t}{g} \operatorname{div}_\eta (q_l \delta p \vec{v})$	<code>cumulated</code> if LHDQLN=.TRUE.
FQLFLUVERTDYN	$\frac{\delta t}{g} q_l \dot{\eta} \frac{\partial p}{\partial \eta}$	<code>cumulated</code> if LHDQLN=.TRUE.
FQLTUR	$\delta t F_{q_l}^{tur}$	<code>cumulated</code> if LHDQLN=.TRUE.
FQLTURCONV	$\delta t F_{q_l}^{tur-conv}$	<code>cumulated</code> if LHDQLN=.TRUE.
FQLTURQNEGAT	$\delta t F_{q_l}^{q<0}$	<code>cumulated</code> if LHDQLN=.TRUE.
VQN0	$\frac{1}{g} q_n \delta p (t=0)$	if LHDQLN=.TRUE.
VQN1	$\frac{1}{g} q_n \delta p (t=NSTEP)$	if LHDQLN=.TRUE.
TQNDIVFLUHOR	$-\frac{\delta t}{g} \operatorname{div}_\eta (q_n \delta p \vec{v})$	<code>cumulated</code> if LHDQLN=.TRUE.
FQNFLUVERTDYN	$\frac{\delta t}{g} q_n \dot{\eta} \frac{\partial p}{\partial \eta}$	<code>cumulated</code> if LHDQLN=.TRUE.
FQNTUR	$\delta t F_{q_n}^{tur}$	<code>cumulated</code> if LHDQLN=.TRUE.
FQNTURCONV	$\delta t F_{q_n}^{tur-conv}$	<code>cumulated</code> if LHDQLN=.TRUE.
FQNTURQNEGAT	$\delta t F_{q_n}^{q<0}$	<code>cumulated</code> if LHDQLN=.TRUE.

4.3 Momentum budget

Budget equation

$$\begin{aligned} \frac{\partial}{\partial t} \left(\frac{1}{g} \frac{\partial p}{\partial \eta} \vec{v} \right) = & -\frac{1}{g} \vec{v} \left[\operatorname{div}_\eta \left(\frac{\partial p}{\partial \eta} \vec{v} \right) + \frac{\partial}{\partial \eta} \left(\dot{\eta} \frac{\partial p}{\partial \eta} \right) \right] - \frac{1}{g} \frac{\partial p}{\partial \eta} (\vec{v} \cdot \vec{\nabla}) \vec{v} - \frac{1}{g} \frac{\partial p}{\partial \eta} f \vec{k} \times \vec{v} \\ & - \frac{1}{g} \frac{\partial p}{\partial \eta} (\vec{\nabla} \Phi + RT \vec{\nabla} \ln p) - \delta_m \frac{\partial F_p \vec{v}}{\partial \eta} - \frac{\partial}{\partial \eta} \left(\vec{F}_v^{tur} + \vec{F}_v^{tur-conv} + \vec{F}_v^{rel} + \vec{F}_v^{meso} \right) \end{aligned}$$

where $\vec{v} = (u, v)$ represents the real wind projected in the local geographic coordinates (u positive towards the East, v positive towards the North) and

- \vec{F}_v^{tur} is the turbulent flux,
- $\vec{F}_v^{tur-conv}$ is the convective transport,
- \vec{F}_v^{rel} is the momentum flux due to gravity wave drag.

In the model (u^{*l} , v^{*l}) are expressed on the transformed sphere. One must, therefore, go back to the real wind (from transformed sphere to real sphere, then modification by the scale factor) before making a rotation, given by

$$\begin{pmatrix} \vec{e} \\ \vec{n} \end{pmatrix} = \begin{pmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{pmatrix} \begin{pmatrix} \vec{i}' \\ \vec{j}' \end{pmatrix}$$

where (\vec{e}, \vec{n}) stands for the local geographical vectors. In ARPEGE $\cos \alpha = \text{GNORDM}$ and $\sin \alpha = \text{GNORDL}$. Likewise for gradients.

Pressure force : the value written on file is

$$-\frac{1}{g} \frac{\partial p}{\partial \eta} \left[f \vec{k} \times \vec{v} + \vec{\nabla} \Phi + RT \vec{\nabla} \ln p \right]$$

File output

All wind components (variables, fluxes, tendencies) are relative to the true geographical sphere.

VUU0	$\frac{1}{g} u \delta p (t=0)$
VVV0	$\frac{1}{g} v \delta p (t=0)$

VUU1	$\frac{1}{g} u \delta p \text{ (t=NSTEP } \delta t)$	
VVV1	$\frac{1}{g} v \delta p \text{ (t=NSTEP } \delta t)$	
TUUDIVFLUHOR	$-\frac{\delta t}{g} u \operatorname{div}_\eta (\delta p \vec{v})$	cumulated
TVVDIVFLUHOR	$-\frac{\delta t}{g} v \operatorname{div}_\eta (\delta p \vec{v})$	cumulated
The wind tendency due to advection of wind by itself is missing !		
TUUFFVGADPSG	$\delta t \left(\frac{f}{g} v \delta p - \frac{\delta p}{g} \left(\frac{\partial \Phi}{\partial x} + RT \frac{\partial \ln p}{\partial x} \right) \right)$	cumulated
TVVFFUGADPSG	$-\delta t \left(\frac{f}{g} u \delta p + \frac{\delta p}{g} \left(\frac{\partial \Phi}{\partial y} + RT \frac{\partial \ln p}{\partial y} \right) \right)$	cumulated
FUUFLUVERTDYN	$\frac{\delta t}{g} u \dot{\eta} \frac{\partial p}{\partial \eta}$	cumulated
FVVFLUVERTDYN	$\frac{\delta t}{g} v \dot{\eta} \frac{\partial p}{\partial \eta}$	cumulated
FUUFLUDUAPLUI	$\delta t \delta_m F_p u$	cumulated
FVVFLUDUAPLUI	$\delta t \delta_m F_p v$	cumulated
FUUTUR	$\delta t F_u^{tur}$	cumulated
FVVTUR	$\delta t F_v^{tur}$	cumulated
FUUTURCONV	$\delta t F_u^{tur-conv}$	cumulated
FVVTURCONV	$\delta t F_v^{tur-conv}$	cumulated
FUUONDEGREL	$\delta t F_u^{rel}$	cumulated
FVVONDEGREL	$\delta t F_v^{rel}$	cumulated
FUUMESO	$\delta t F_u^{meso}$	cumulated
FVVMESO	$\delta t F_v^{meso}$	cumulated

4.4 Kinetic energy budget

Budget equation

$$\begin{aligned} \frac{\partial}{\partial t} \left(\frac{1}{g} k \frac{\partial p}{\partial \eta} \right) = & -\frac{1}{g} k \operatorname{div}_\eta \left(\vec{v} \frac{\partial p}{\partial \eta} \right) - \frac{1}{g} \frac{\partial}{\partial \eta} \left(k \dot{\eta} \frac{\partial p}{\partial \eta} \right) - \frac{1}{g} \frac{\partial p}{\partial \eta} \vec{v} \cdot \vec{\nabla} k \\ & - \frac{1}{g} \frac{\partial p}{\partial \eta} \vec{v} \left(\vec{\nabla} \Phi + RT \vec{\nabla} \ln p \right) - \delta_m \frac{\partial k F_p}{\partial \eta} - \vec{v} \frac{\partial}{\partial \eta} \left(\vec{F}_v^{tur} + \vec{F}_v^{tur-conv} + \vec{F}_v^{rel} + \vec{F}_v^{meso} \right) \end{aligned}$$

The work due to pressure gradient force is precisely known by LAGGRAD. However, the advection term cannot be diagnosed at initial time.

File output

VKK0	$\frac{1}{g} k \delta p \text{ (t=0)}$	
VKK1	$\frac{1}{g} k \delta p \text{ (t=NSTEP } \delta t)$	
TKKDIVFLUHOR	$-\frac{\delta t}{g} k \operatorname{div}_\eta (\delta p \vec{v})$	cumulated (the advection term is missing !)
TKKCONVERSI1	$-\frac{\delta t}{g} \delta p \vec{v} \left(\vec{\nabla} \Phi + RT \vec{\nabla} \ln p \right)$	cumulated
FKKFLUVERTDYN	$\frac{\delta t}{g} k \dot{\eta} \frac{\partial p}{\partial \eta}$	cumulated

FKKFLUDUAPLUI	$\delta t \delta_m k F_p$	cumulated
TKKDISSIPTUR	$-\delta t \vec{v} \delta \vec{F}_v^{tur}$	cumulated
TKKDISSIPCONV	$-\delta t \vec{v} \delta \vec{F}_v^{tur-conv}$	cumulated
TKKDISSIPGREL	$-\delta t \vec{v} \delta \vec{F}_v^{rel}$	cumulated
TKKDISSIPMESO	$-\delta t \vec{v} \delta \vec{F}_v^{meso}$	cumulated

4.5 Thermal energy budget

Two types of thermal energy ($c_p T$) equations are used in ARPEGE : the DDH use the budget type (4.3), the CPTEND routine uses the eulerien type in s (4.2). The corresponding lagrangian equation in T is given for information in (4.1).

Lagrangian equation in T

$$\begin{aligned}
 r_\eta c_p \frac{dT}{dt} &= r_\eta RT \frac{\omega}{p} + \frac{\partial F_{c_p T}}{\partial \eta} - L_{v>l}(T) \frac{\partial F_c^l}{\partial \eta} - L_{v>n}(T) \frac{\partial F_c^n}{\partial \eta} \\
 &+ F_p^l \frac{\partial}{\partial \eta} \{ T [c_l - c_{p_a}(1 - \delta_m)] \} + F_p^n \frac{\partial}{\partial \eta} \{ T [c_n - c_{p_a}(1 - \delta_m)] \} \\
 &- T \left[(c_{p_v} - c_{p_a}) \frac{\partial (F_{qv}^{tur} + F_{qv}^{tur-conv})}{\partial \eta} + (c_l - c_{p_a}) \frac{\partial (F_{ql}^{tur} + F_{ql}^{tur-conv})}{\partial \eta} + (c_n - c_{p_a}) \frac{\partial (F_{qn}^{tur} + F_{qn}^{tur-conv})}{\partial \eta} \right] \\
 &+ \delta_m F_p \frac{\partial (\Phi + \frac{u^2 + v^2}{2})}{\partial \eta} - \vec{v} \cdot \frac{\partial \vec{F}_v^{phys}}{\partial \eta}
 \end{aligned} \tag{4.1}$$

Eulerian equation in $s = c_p T + \Phi + \frac{u^2 + v^2}{2}$

$$\begin{aligned}
 r_\eta \frac{\partial s}{\partial t} &= -r_\eta (\vec{v} \cdot \vec{\nabla}) s - r_\eta \dot{\eta} \frac{\partial s}{\partial \eta} + r_\eta RT \frac{\omega}{p} \\
 &+ \frac{\partial}{\partial \eta} \left\{ F_{c_p T} + F_{c_p T_{prec}} + F_p^l T [c_l - c_{p_a}(1 - \delta_m)] + F_p^n T [c_n - c_{p_a}(1 - \delta_m)] - \delta_m c_p T F_p \right\} \\
 &+ \delta_m F_p \frac{\partial s}{\partial \eta}
 \end{aligned} \tag{4.2}$$

One assumes $\frac{\partial \Phi}{\partial t} = 0$.

Budget equation

$$\begin{aligned}
 \frac{\partial}{\partial t} (r_\eta c_p T) &= -\text{div}_\eta (r_\eta c_p T \vec{v}) - \frac{\partial}{\partial \eta} (r_\eta c_p T \dot{\eta}) + r_\eta RT \frac{\omega}{p} \\
 &+ \frac{\partial}{\partial \eta} \left\{ F_{c_p T} + F_{c_p T_{prec}} + F_p^l T [c_l - c_{p_a}(1 - \delta_m)] + F_p^n T [c_n - c_{p_a}(1 - \delta_m)] \right\} \\
 &+ \delta_m F_p \frac{\partial (\Phi + \frac{u^2 + v^2}{2})}{\partial \eta} - \vec{v} \cdot \frac{\partial \vec{F}_v^{phys}}{\partial \eta}
 \end{aligned} \tag{4.3}$$

where

- $r_\eta = -\frac{1}{g} \frac{\partial p}{\partial \eta}$.
- $F_{c_p T} = F_{c_p T}^{sol} + F_{c_p T}^{ther} + F_{c_p T}^{meso} + F_s^{tur} + F_s^{tur-conv}$.
- $F_s^{tur-conv}$ is the subgrid-scale transport of dry static energy $s = c_p T + \Phi$, due to deep convection.
- $F_{c_p T_{prec}} = F_{c_p T_{prec}}^l + F_{c_p T_{prec}}^n$.
- $F_{c_p T_{prec}}^l = F_{c_p T_{prec}}^{conv-l} + F_{c_p T_{prec}}^{stra-l}$.
- $F_{c_p T_{prec}}^n = F_{c_p T_{prec}}^{conv-n} + F_{c_p T_{prec}}^{stra-n}$.

- $F_{c_p T_{prec}}^{conv-l} = - [L_{v>l}(T) + (c_l - c_{p_v})T] F_c^{conv-l}$.
- $F_{c_p T_{prec}}^{conv-n} = - [L_{v>n}(T) + (c_n - c_{p_v})T] F_c^{conv-n}$.
- $F_{c_p T_{prec}}^{stra-l} = - [L_{v>l}(T) + (c_l - c_{p_v})T] F_c^{stra-l}$.
- $F_{c_p T_{prec}}^{stra-n} = - [L_{v>n}(T) + (c_n - c_{p_v})T] F_c^{stra-n}$.
- $F_{c_p T}^{sol}$ et $F_{c_p T}^{ther}$ are the solar and infra-red fluxes.
- $\vec{F}_v^{phys} = \vec{F}_v^{tur} + \vec{F}_v^{tur-conv} + \vec{F}_v^{rel} + \vec{F}_v^{meso}$.
- $c_p = c_{p_a} q_a + c_{p_v} q_v + c_l q_l + c_n q_n$.

File output

VCT0	$\frac{1}{g} c_p T \delta p$ (t=0)	
VCT1	$\frac{1}{g} c_p T \delta p$ (t= NSTEP δt)	
TCTDIVFLUHOR	$-\frac{\delta t}{g} \text{div}_\eta (c_p T \delta p \vec{v})$	cumulated
TCTCONVERSI2	$\frac{\delta t}{g} RT \delta p (\omega/p)$	cumulated
TCTCONVERSI3	$-\delta_m \delta t F_p \delta \Phi$	cumulated
FCTFLUVERTDYN	$\frac{\delta t}{g} c_p T \dot{\eta} \frac{\partial p}{\partial \eta}$	cumulated
FCTTUR	$\delta t F_s^{tur}$	cumulated
FCTTURCONV	$\delta t F_s^{tur-conv}$	cumulated
FCTRAYSOL1	$\delta t F_{c_p T}^{sol}$	cumulated
FCTRAYER1	$\delta t F_{c_p T}^{ther}$	cumulated
FCTMESO	$\delta t F_{c_p T}^{meso}$	cumulated
FCTPRECISTL	$-\delta t F_{c_p T_{prec}}^{stra-l}$	cumulated
FCTPRECISTN	$-\delta t F_{c_p T_{prec}}^{stra-n}$	cumulated
FCTPRECICOL	$-\delta t F_{c_p T_{prec}}^{conv-l}$	cumulated
FCTPRECICON	$-\delta t F_{c_p T_{prec}}^{conv-n}$	cumulated
FCTPRECCSCOL	$-\delta t F_p^{conv-l} T [c_l - c_{p_a} (1 - \delta_m)]$	cumulated
FCTPRECCSCON	$-\delta t F_p^{conv-n} T [c_n - c_{p_a} (1 - \delta_m)]$	cumulated
FCTPRECCSSTL	$-\delta t F_p^{stra-l} T [c_l - c_{p_a} (1 - \delta_m)]$	cumulated
FCTPRECCSSTN	$-\delta t F_p^{stra-n} T [c_n - c_{p_a} (1 - \delta_m)]$	cumulated

4.6 Angular momentum budget

The angular momentum is defined by

$$\vec{M} = \vec{r} \times (\vec{\Omega} \times \vec{r} + \vec{v})$$

where \vec{r} stands for the position vector, \vec{v} the horizontal wind and $\vec{\Omega}$ the Earth pulsation. In Cartesian coordinate, the kinetic moment reads

$$\begin{aligned}
 \vec{M} &= M_1 \vec{i} + M_2 \vec{j} + M_3 \vec{k} \\
 &= (av \sin \lambda - a(u + a\Omega \cos \theta) \sin \theta \cos \lambda) \vec{i} \\
 &\quad + (-av \cos \lambda - a(u + a\Omega \cos \theta) \sin \theta \sin \lambda) \vec{j} \\
 &\quad + a(u + a\Omega \cos \theta) \cos \theta \vec{k}
 \end{aligned}$$

with

- $(\vec{i}, \vec{j}, \vec{k})$: \vec{k} along the earth rotation axis, \vec{i} et \vec{j} in the equatorial plane. This base does not rotate with the earth : the position of the Greenwich meridian at 00 UTC on the day of the beginning of the integration of the model sets \vec{i} . \vec{j} is got by an equatorial rotation of 90° towards the East of \vec{i} , and of $\vec{k} = \vec{i} \times \vec{j}$.
- u and v zonal and meridional wind.
- a earth radius.
- θ and λ latitude and longitude.

Budget equation

$$\frac{\partial}{\partial t} (r_\eta \vec{M}) = -\vec{M} \left[\text{div}_\eta (r_\eta \vec{v}_a) + \frac{\partial}{\partial \eta} (r_\eta \dot{\eta}) \right] - r_\eta (\vec{v}_a \cdot \vec{\nabla}) \vec{M} + \delta_m \vec{M} \frac{\partial F_p}{\partial \eta} + r_\eta \vec{r} \times \vec{\alpha}$$

avec

- $r_\eta = -\frac{1}{g} \frac{\partial p}{\partial \eta}$
- $\vec{v}_a = \vec{\Omega} \times \vec{r} + \vec{v}$
- $\vec{\alpha} = \vec{\Omega} \times (\vec{\Omega} \times \vec{r}) - \frac{1}{r_\eta} \left[\Phi \vec{\nabla} r_\eta + \frac{1}{g} \frac{\partial}{\partial \eta} (\Phi \vec{\nabla} p) \right] - \vec{\nabla} \Phi + \frac{1}{r_\eta} \left[\frac{\partial \vec{F}_v}{\partial \eta} + \delta_m F_p \frac{\partial \vec{v}}{\partial \eta} \right]$
- $\vec{F}_v = \vec{F}_v^{tur} + \vec{F}_v^{tur-conv} + \vec{F}_v^{rel}$

File output

VA10	$\frac{\delta p}{g} M_1 (t=0)$	
VA20	$\frac{\delta p}{g} M_2 (t=0)$	
VA30	$\frac{\delta p}{g} M_3 (t=0)$	
VA11	$\frac{\delta p}{g} M_1 (t=\text{NSTEP } \delta t)$	
VA21	$\frac{\delta p}{g} M_2 (t=\text{NSTEP } \delta t)$	
VA31	$\frac{\delta p}{g} M_3 (t=\text{NSTEP } \delta t)$	
TA1DIVFLUHOR	$-\frac{\delta t}{g} M_1 \left[\text{div}_\eta (\vec{v} \delta p) + a \Omega \cos \theta \delta B \frac{\partial \pi}{\partial x} \right]$	cumulated
TA2DIVFLUHOR	$-\frac{\delta t}{g} M_2 \left[\text{div}_\eta (\vec{v} \delta p) + a \Omega \cos \theta \delta B \frac{\partial \pi}{\partial x} \right]$	cumulated
TA3DIVFLUHOR	$-\frac{\delta t}{g} M_3 \left[\text{div}_\eta (\vec{v} \delta p) + a \Omega \cos \theta \delta B \frac{\partial \pi}{\partial x} \right]$	cumulated
FA1FLUVERTDYN	$\delta t \frac{\delta p}{g} M_1 \dot{\eta}$	cumulated
FA2FLUVERTDYN	$\delta t \frac{\delta p}{g} M_2 \dot{\eta}$	cumulated
FA3FLUVERTDYN	$\delta t \frac{\delta p}{g} M_3 \dot{\eta}$	cumulated
TA1ADJUST	$\frac{\delta t}{g} a \left[- \left(\delta p \frac{\partial RT}{\partial y} + RT \delta B \frac{\partial \Phi}{\partial y} \right) \sin(\lambda + \Omega t) + \left(\delta p \frac{\partial RT}{\partial x} + RT \delta B \frac{\partial \Phi}{\partial x} \right) \cos(\lambda + \Omega t) \sin \theta \right]$	cumulated
TA2ADJUST	$\frac{\delta t}{g} a \left[- \left(\delta p \frac{\partial RT}{\partial y} + RT \delta B \frac{\partial \Phi}{\partial y} \right) \cos(\lambda + \Omega t) + \left(\delta p \frac{\partial RT}{\partial x} + RT \delta B \frac{\partial \Phi}{\partial x} \right) \sin(\lambda + \Omega t) \sin \theta \right]$	cumulated
TA3ADJUST	$-\frac{\delta t}{g} a \left(\delta p \frac{\partial RT}{\partial x} + RT \delta B \frac{\partial \Phi}{\partial x} \right) \cos \theta$	cumulated
TA1NONAX	$\delta t \frac{\delta p}{g} a^2 \Omega^2 \sin \theta \cos \theta \sin(\lambda + \Omega t)$	cumulated
TA2NONAX	$\delta t \frac{\delta p}{g} a^2 \Omega^2 \sin \theta \cos \theta \cos(\lambda + \Omega t)$	cumulated
FA1GRAV	$-\frac{\delta p}{g} a \left[\sin \theta \cos(\lambda + \Omega t) \frac{\partial \Phi}{\partial x} + \sin(\lambda + \Omega t) \frac{\partial \Phi}{\partial y} \right]$	cumulated
FA2GRAV	$-\frac{\delta p}{g} a \left[\sin \theta \sin(\lambda + \Omega t) \frac{\partial \Phi}{\partial x} + \cos(\lambda + \Omega t) \frac{\partial \Phi}{\partial y} \right]$	cumulated

FA3GRAV	$\frac{\delta p}{g} a \cos \theta \frac{\partial \Phi}{\partial x}$	cumulated
FA1FLUDUAPLUI	$\delta t \delta_m F_p M_1$	cumulated
FA2FLUDUAPLUI	$\delta t \delta_m F_p M_2$	cumulated
FA3FLUDUAPLUI	$\delta t \delta_m F_p M_3$	cumulated
TA1TUR	$\delta t a [F_v^{tur} \sin(\lambda + \Omega t) - F_u^{tur} \sin \theta \cos(\lambda + \Omega t)]$	cumulated
TA2TUR	$\delta t a [F_v^{tur} \cos(\lambda + \Omega t) - F_u^{tur} \sin \theta \sin(\lambda + \Omega t)]$	cumulated
TA3TUR	$\delta t a F_u^{tur} \cos \theta$	cumulated
TA1TURCONV	$\delta t a [F_v^{tur-conv} \sin(\lambda + \Omega t) - F_u^{tur-conv} \sin \theta \cos(\lambda + \Omega t)]$	cumulated
TA2TURCONV	$\delta t a [F_v^{tur-conv} \cos(\lambda + \Omega t) - F_u^{tur-conv} \sin \theta \sin(\lambda + \Omega t)]$	cumulated
TA3TURCONV	$\delta t a F_u^{tur-conv} \cos \theta$	cumulated
TA1ONDEGREL	$\delta t a [F_v^{rel} \sin(\lambda + \Omega t) - F_u^{rel} \sin \theta \cos(\lambda + \Omega t)]$	cumulated
TA2ONDEGREL	$\delta t a [F_v^{rel} \cos(\lambda + \Omega t) - F_u^{rel} \sin \theta \sin(\lambda + \Omega t)]$	cumulated
TA3ONDEGREL	$\delta t a F_u^{rel} \cos \theta$	cumulated

Listing output

When an output on listing is requested, what is printed is the intersection position of \vec{M} with the earth surface, in latitude and in longitude values :

$$\begin{aligned} \cos \theta_0 &= \frac{M_3}{\sqrt{M_1^2 + M_2^2 + M_3^2}} \\ \cos(\lambda_0 + \Omega t) &= \frac{M_1}{\sqrt{M_1^2 + M_2^2}} \end{aligned}$$

4.7 Entropy budget

Budget equation

$$\begin{aligned} \frac{\partial}{\partial t} (r_\eta s) &= -\text{div}_\eta (r_\eta s \vec{v}) - \frac{\partial}{\partial \eta} (r_\eta s \dot{\eta}) + \frac{\partial}{\partial \eta} [s_l (F_p^{conv-l} + F_p^{stra-l}) + s_g (F_p^{conv-n} + F_p^{stra-n})] \\ &\quad - \frac{1}{T} \vec{v} \cdot \frac{\partial}{\partial \eta} (\vec{F}_v^{tur} + \vec{F}_v^{tur-conv} + \vec{F}_v^{rel}) - (s_v - s_a + c_{pv} - c_{pa}) \frac{\partial}{\partial \eta} (F_q^{tur} + F_q^{tur-conv}) \\ &\quad + \frac{1}{T} (F_{cpT}^{sol} + F_{cpT}^{ther} + F_{cpT}^{tur} + F_{cpT}^{tur-conv}) - (1 - \delta_m) \left[s_a \frac{\partial F_p}{\partial \eta} + \frac{1}{T} c_{pa} \frac{\partial T}{\partial \eta} F_p \right] + \delta_m \frac{1}{T} F_p \frac{\partial \Phi}{\partial \eta} \end{aligned}$$

where

- $r_\eta = -\frac{1}{g} \frac{\partial p}{\partial \eta}$.
- $s = s_a + (s_v - s_a) q_v$.
- $s_a = c_{pa} \ln(\frac{T}{T_0}) - R_a \ln(\frac{p_a}{p_0}) + s_a^0$.
- $s_v = c_{pv} \ln(\frac{T}{T_0}) - R_v \ln(\frac{p_v}{p_0}) + s_v^0$.
- $s_l = c_w \ln(\frac{T}{T_0}) + s_l^0$.
- $s_g = c_g \ln(\frac{T}{T_0}) + s_g^0$.
- $s_a^0 = 6775 \text{ J kg}^{-1} \text{ K}^{-1}$.
- $s_v^0 = 10320 \text{ J kg}^{-1} \text{ K}^{-1}$.
- $s_l^0 = 3517 \text{ J kg}^{-1} \text{ K}^{-1}$.
- $s_g^0 = 2296 \text{ J kg}^{-1} \text{ K}^{-1}$.

The horizontal divergence term is computed as

$$\text{div}_\eta (r_\eta s \vec{v}) = s \text{div}_\eta (r_\eta \vec{v}) + r_\eta \vec{v} \cdot \vec{\nabla} s$$

where

$$\begin{aligned} & \vec{v} \cdot \vec{\nabla} s = (s_v - s_a) \vec{v} \cdot \vec{\nabla} q_v + \vec{v} \cdot \left[c_p \vec{\nabla} \ln T - R \vec{\nabla} \ln p \right]. \\ & \vec{v} \cdot \vec{\nabla} \ln p = \text{RTGR } \vec{v} \cdot \vec{\nabla} \pi. \end{aligned}$$

File output

VSS0	$\frac{1}{g} s \delta p$ (t=0)	
VSS1	$\frac{1}{g} s \delta p$ (t=NSTEP Δt)	
TSSDIVFLUHOR	$-\frac{\delta t}{g} \{ s \text{div}_\eta (\vec{v} \delta p) + \delta p [(s_v - s_a) \vec{v} \cdot \vec{\nabla} q_v + \frac{1}{T} c_p \vec{v} \cdot \vec{\nabla} T - R \text{ZRTGR } \vec{v} \cdot \vec{\nabla} \pi] \}$	cumulated
TSSPRECICT	$\delta t (1 - \delta_m) [s_a \delta F_p + c_{p_a} F_p \frac{\delta T}{T}] - \delta_m \delta t \frac{1}{T} F_p \delta \Phi$	cumulated
FSSFLUVERTDYN	$\frac{\delta t}{g} s \dot{\eta} \delta p$	cumulated
FSSPRECISS	$\delta t [s_l (F_p^{\text{conv}-l} + F_p^{\text{stra}-l}) + s_g (F_p^{\text{conv}-n} + F_p^{\text{stra}-n})]$	cumulated
TSSDISSIPTUR	$\frac{\delta t}{T} \vec{v} \cdot \delta \vec{F}_v^{\text{tur}}$	cumulated
TSSDISSIPCONV	$\frac{\delta t}{T} \vec{v} \cdot \delta \vec{F}_v^{\text{tur}-\text{conv}}$	cumulated
TSSDISSIPGREL	$\frac{\delta t}{T} \vec{v} \cdot \delta \vec{F}_v^{\text{rel}}$	cumulated
TSTURQVTOT	$\delta t (s_v - s_a + c_{p_v} - c_{p_a}) (\delta F_q^{\text{tur}} + \delta F_q^{\text{tur}-\text{conv}})$	cumulated
TSSRAY1	$-\frac{\delta t}{T} (\delta F_{cpT}^{\text{sol}} + F_{cpT}^{\text{ther}})$	cumulated
TSTURCTTOT	$-\frac{\delta t}{T} (\delta F_{cpT}^{\text{tur}} + F_{cpT}^{\text{tur}-\text{conv}})$	cumulated

4.8 Potential energy budget

Budget equation

$$\begin{aligned} -\frac{1}{g} \frac{\partial p}{\partial \eta} \vec{v} \cdot (\vec{\nabla} \Phi + RT \vec{\nabla} \ln p) &= -RT \left[\frac{\omega}{gp} + \delta_m \frac{F_p}{p} \right] \frac{\partial p}{\partial \eta} - \frac{1}{g} \text{div}_\eta \left(\Phi \frac{\partial p}{\partial \eta} \vec{v} \right) \\ &\quad - \frac{1}{g} \frac{\partial}{\partial \eta} \left[\Phi \left(\frac{\partial p}{\partial t} + \dot{\eta} \frac{\partial p}{\partial \eta} + \delta_m g F_p \right) \right] \end{aligned}$$

File output

Three of these budget terms have already been mentioned, see « kinetic energy budget » et « thermal energy budget ». Is added

VEP0	$\frac{1}{g} \Phi \delta p$ (t=0)	
VEP1	$\frac{1}{g} \Phi \delta p$ (t=NSTEP δt)	
TEPDIVFLUHOR	$-\frac{\delta t}{g} \text{div}_\eta (\Phi \delta p \vec{v})$	cumulated

$$\text{FEPCONVERSIFL} \quad -\frac{\delta t}{g} \Phi_{\vec{\ell}} \sum_{k=1}^{\ell} \text{div}_{\eta} (\delta p \vec{v}) \quad \text{cumulated}$$

4.9 Surface budgets

The DDH tool does not produce a closed soil budget, rather some surface fluxes or variables multiplied by land/sea mask `PITM` (1 on land, 0 at sea). The surface occupied by these points in each domain will also be computed. This operation makes it possible, to bring back the variable mean and the mean of fluxes to the sole surface of the continents.

Budget equation

— Surface temperature :

$$\frac{\partial c_{ms} T_s}{\partial t} = \delta_{terre} \left\{ F_{cpT_L}^{sol} + F_{cpT_L}^{ther} + L_v(T_s) E_l + L_n(T_s) E_n + F_{csa} - F_{csp} - L_{fonte} F_{fonte} \right\}$$

where c_{ms} is the surfacic calorific capacity of the surface layer. A constant value is used here : `1/HSOL`.

— Deep temperature :

$$\frac{\partial c_{mp} T_p}{\partial t} = \delta_{terre} F_{csp}$$

where c_{mp} is the surfacic calorific capacity of the deep layer. A constant value is used here : `RTINER/HSOL`.

— Surface water content :

$$\frac{\partial w_s}{\partial t} = \delta_{terre} \left\{ F_p^{conv-l} + F_p^{stra-l} + E_l - F_{perco} + F_{fonte} - F_{ruiss} \right\}$$

— Deep water content :

$$\frac{\partial w_p}{\partial t} = \delta_{terre} \left\{ F_{perco} - F_{ruisp} \right\}$$

— Surface snow content :

$$\frac{\partial w_n}{\partial t} = \delta_{neige} \delta_{terre} \left\{ F_p^{conv-n} + F_p^{stra-n} + E_n - F_{fonte} \right\}$$

File output

The fields are written on 3 file articles :

— Variables at $t = 0$: article `Sxx_0`

$$\frac{1}{S_D} \sum_{(j,k) \in \mathcal{D}} \text{PITM} \sigma_{j,k} \quad \text{Land fraction}$$

$c_{ms} T_s$ Surface layer : energy

$c_{mp} T_p$ Deep layer : energy

w_s Surface layer : water content (surfacic mass)

w_n Surface layer : snow content (surfacic mass)

w_p Deep layer : water content (surfacic mass)

— Variables at $t = NSTEP \times \delta t$: article `Sxx_1`

The same articles as above.

— Flux cumulated from 0 to `NSTEP` : articles `G01, ..., G17`

G01	$F_{cpT} \tilde{L}^{sol}$	Solar radiation
G02	$F_{cpT} \tilde{L}^{ther}$	Infra-red radiation
G03	$L_v(T_s)E_l$	Latent heat (water)
G04	$L_n(T_s)E_n$	Latent heat (snow)
G05	F_{csa}	Sensible heat at surface
G06	F_{csp}	Sensible heat between surface and deep layer
G07	$L_{fonte} F_{fonte}$	Surface flux due to melting Flux de chaleur en surface lié à la fonte de neige
G08	F_p^{stra-l}	Precipitation : resolved, rain
G09	F_p^{stra-n}	Precipitation : resolved, snow
G10	F_p^{conv-l}	Precipitation : subgrid-scale, rain
G11	F_p^{conv-n}	Precipitation : subgrid-scale, snow
G12	E_l	Surface evaporation (water)
G13	E_n	Surface evaporation (snow)
G14	F_{perco}	Percolation from surface to deep
G15	F_{fonte}	Snow melting
G16	F_{ruiss}	Surface run-off
G17	F_{ruisp}	Deep run-off

4.10 Instantaneous diagnostics

4.10.1 Relative humidity

VHR0	$\frac{1}{g} H_r \delta p$ (t=0)
VHR1	$\frac{1}{g} H_r \delta p$ (t=NSTEP δt)

4.10.2 Cloudiness

VNT0	$\frac{1}{g} n_t \delta p$ (t=0)
VNT1	$\frac{1}{g} n_t \delta p$ (t=NSTEP δt)

4.10.3 Vertical velocity

VOM0	$\frac{1}{g} \omega \delta p$ (t=0)
VOM1	$\frac{1}{g} \omega \delta p$ (t=NSTEP δt)

4.10.4 Passive variables

The passive variables are diagnosed if LHDHKS is true and if NFPASS (number of passive variables in the model) is higher or equal to 1.

$$\begin{aligned} V_{xx0} & \quad \frac{1}{g} v_{xx} \delta p \text{ (t=0)} \\ V_{xx1} & \quad \frac{1}{g} v_{xx} \delta p \text{ (t=NSTEP } \delta t) \end{aligned}$$

where xx is between 1 and NFPASS.

4.10.5 Free style variables

SVGFS01	2 m temperature
SVGFS02	specific water vapour at 2 m temperature
SVGFS03	u at 2 m level
SVGFS04	v at 2 m level
SVGFS05	$g z$ orography
SVGFS06	$g z_0$, where z_0 is the dynamical roughness
SVGFS07	$g z_0$, where z_0 is the thermal roughness
SVGFS08	albedo
SVGFS09	boundary layer height
SVGFS10	surface temperature T_s

These arrays are initialized in cphddh.F90.

4.10.6 Free style fluxes

SFGFS01	surface downward solar radiation
SFGFS02	surface downward thermal radiation
SFGFS03	surface sensible flux
SFGFS04	surface latent heat flux liquid
SFGFS05	surface latent heat flux solid

These arrays are initialized in cphddh.F90.

4.11 Cumulated mass

To make possible future conversion of tendencies and fluxes (extensive) in intensive values, the following value is diagnosed :

PPP	$\frac{1}{g} \delta t \delta p$	cumulated
-----	---------------------------------	-----------

Chapitre 5

Budgets and diagnostics, AROME model

5.1 Balance equations

5.1.1 Momentum

FUUTUR	F_u^{tur}	vertical turbulence flux of u velocity component
FVVTUR	F_v^{tur}	vertical turbulence flux of v velocity component
FVWTUR	F_w^{tur}	vertical turbulence flux of w velocity component

5.1.2 Turbulence kinetic energy

FTETURB	F_{tke}^{tur}	turbulent flux of turbulent kinetic energy
FTEDYPRO	$F_{tke}^{tur-prod-dyn}$	dynamic production of turbulent kinetic energy
FTETERMPRO	$F_{tke}^{tur-prod-term}$	thermic production of turbulent kinetic energy
FTEDISS	$F_{tke}^{tur-diss}$	dissipation of turbulent kinetic energy

5.1.3 Thermal energy

FCTNEGC1	F_{cpT}^{negc1}	correction of negative specific ratios after advection
FCTCDEPI	F_{cpT}^{cdepi}	adjustment of water vapour, cloud water and cloud ice
FCTVCONV	F_{cpT}^{conv}	convection flux of thermal energy
FCTVTURB	F_{cpT}^{tur}	vertical turbulent flux of thermal energy
FCTDISSTUR	$F_{cpT}^{tur-diss}$	dissipation of turbulent kinetic energy
FCTNEGC	F_{cpT}^{negc}	correction of negative specific ratios after turbulence
FCTHENUI	F_{cpT}^{henuv}	heterogeneous nucleation of ice
FCTHON	F_{cpT}^{honl}	homogeneous nucleation of ice
FCTSFR	F_{cpT}^{sfr}	spontaneous freezing
FCTDEPS	F_{cpT}^{deps}	deposition on snow
FCTDEPG	F_{cpT}^{depg}	deposition on graupel
FCTREVA	F_{cpT}^{reva}	rain evaporation
FCTRIM	F_{cpT}^{rim}	riming by cloud droplets
FCTACCS	F_{cpT}^{accs}	collection of raindrops and snow on graupel
FCTCFRZ	F_{cpT}^{cfrz}	contact freezing of rain
FCTWETG	F_{cpT}^{wetg}	wet growth of graupel
FCTDRYG	F_{cpT}^{dryg}	dry growth of graupel
FCTMLTG	F_{cpT}^{mltg}	melting of graupel
FCTMLTI	F_{cpT}^{mlti}	melting of cloud ice
FCTBERFI	F_{cpT}^{berfi}	Bergeron-Findeisen effect
FCTRAYSOL1	$F_{cpT}^{raysol1}$	solar radiation
FCTRAYER1	$F_{cpT}^{rayter1}$	earth radiation

5.1.4 Water vapour

FQVNEGC1	F_{qv}^{negc1}	correction of negative specific ratios after advection
FQVDEPI	F_{qv}^{depi}	adjustment of water vapour, cloud water and cloud ice
FQVVCONV	F_{qv}^{conv}	convection flux of water vapour
FQVVTURB	F_{qv}^{tur}	vertical turbulent flux of water vapour
FQVNEGC	F_{qv}^{negc}	correction of negative specific ratios after turbulence
FQVHENUI	F_{qv}^{henu}	heterogeneous nucleation of ice
FQVDEPS	F_{qv}^{deps}	deposition on snow
FQVDEPG	F_{qv}^{depg}	deposition on groupel
FQVREVA	F_{qv}^{reva}	rain evaporation

5.1.5 Cloud water

FQLNEGC1	F_{ql}^{negc1}	correction of negative specific ratios after advection
FQLCDEPI	F_{ql}^{cddep}	adjustment of water vapour, cloud water and cloud ice
FQLVCONV	F_{ql}^{conv}	convection flux of cloud water
FQLVTURB	F_{ql}^{tur}	vertical turbulent flux of cloud water
FQLNEGC	F_{ql}^{negc}	correction of negative specific ratios after turbulence
FQLHON	F_{ql}^{hon}	homogeneous nucleation of ice
FQLAUTO	F_{ql}^{autor}	auto-conversion of cloud water
FQLACCR	F_{ql}^{accr}	accretion of cloud water on rain
FQLRIMS	F_{ql}^{rim}	riming by cloud droplets
FQLWETG	F_{ql}^{wetg}	wet growth of graupel
FQLDRYG	F_{ql}^{dryg}	dry growth of graupel
FQLMLTI	F_{ql}^{mlti}	melting of cloud ice
FQLBERFI	F_{ql}^{berfi}	Bergeron-Findeisen effect

5.1.6 Rain

FQRNEGC	F_{qr}^{negc}	correction of negative specific ratios after advection
FQRSEDI	F_{rp}	sedimentation
FQRSFR	F_{qr}^{sfrz}	spontaneous freezing of rain
FQRAUTO	F_{qr}^{autor}	auto-conversion of cloud water
FQRACCL	F_{qr}^{accr}	accretion of cloud water on rain
FQRREVA	F_{qr}^{reva}	rain evaporation
FQRACCS	F_{qr}^{accs}	collection of raindrops on graupel
FQRCFRZ	F_{qr}^{cfrz}	contact freezing of rain
FQRWETG	F_{qr}^{wetg}	wet growth of graupel
FQRDRYG	F_{qr}^{dryg}	dry growth of graupel
FQRMLTG	F_{qr}^{mltg}	melting of graupel

5.1.7 Cloud ice

FQINEGC1	$F_{q_i}^{negc1}$	correction of negative specific ratios after advection
FQICDEPI	$F_{q_i}^{cdep_i}$	adjustment of water vapour, cloud water and cloud ice
FQICONV	$F_{q_i}^{conv}$	convection flux of cloud ice
FQITURB	$F_{q_i}^{tur}$	vertical turbulent flux of cloud ice
FQINEGC	$F_{q_i}^{negc}$	correction of negative specific ratios after turbulence
FQISED1	F_{ip}	sedimentation
FQIHENU	$F_{q_i}^{henu}$	heterogeneous nucleation of ice
FQIHON	$F_{q_i}^{hon}$	homogeneous nucleation of ice
FQIAGGS	$F_{q_i}^{agg}$	collection of ice on snow
FQIAUTS	$F_{q_i}^{autoi}$	auto-conversion of ice to snow
FQICFRZ	$F_{q_i}^{cfrz}$	contact freezing of rain
FQIWETG	$F_{q_i}^{wetg}$	wet growth of graupel
FQIDRYG	$F_{q_i}^{dryg}$	dry growth of graupel
FQIMLT	$F_{q_i}^{mlti}$	melting of cloud ice
FQIBERFI	$F_{q_i}^{berfi}$	Bergeron-Findeisen effect

5.1.8 Snow

FQSNEGC	$F_{q_s}^{negc}$	correction of negative specific ratios after advection
FQSSEDI	F_{sp}	sedimentation
FQSDEPS	$F_{q_s}^{dep}$	deposition on snow
FQSAGGS	$F_{q_s}^{agg}$	collection of ice on snow
FQSAUTS	$F_{q_s}^{autoi}$	auto-conversion of ice to snow
FQSRIM	$F_{q_s}^{rim}$	riming by cloud droplets
FQSACC	$F_{q_s}^{accs}$	collection of raindrops and snow on graupel
FQSCMEL	$F_{q_s}^{cmel}$	melting of aggregates
FQSWETG	$F_{q_s}^{wetg}$	wet growth of graupel
FQSDRYG	$F_{q_s}^{dryg}$	dry growth of graupel

5.1.9 Graupel

FQGNCG	$F_{q_g}^{negc}$	correction of negative specific ratios after advection
FQGSEDI	F_{gp}	sedimentation
FQGSFR	$F_{q_g}^{sfr}$	spontaneous freezing
FQGDEPG	$F_{q_g}^{dep}$	deposition on graupel
FQGRIM	$F_{q_g}^{rim}$	riming by cloud droplets
FQGACC	$F_{q_g}^{accs}$	collection of raindrops and snow on graupel
FQGCMEI	$F_{q_g}^{cmel}$	melting of aggregates
FQGCFRZ	$F_{q_g}^{cfrz}$	contact freezing of rain
FQGWETG	$F_{q_g}^{wetg}$	wet growth of graupel
FQGDRYG	$F_{q_g}^{dryg}$	dry growth of graupel
FQGMLT	$F_{q_g}^{mltg}$	melting of graupel

5.2 Common Dynamics-Physics Interface CDPI

FQVPL1	P'_l	Pseudo flux due to condensation
FQVPI1	P'_i	Pseudo flux due to sublimation
FQLPL2	P''_l	Pseudo flux due to evaporation of rain
FQIPI2	P''_i	Pseudo flux due to conversion of cloud ice to snow and graupel
FQRPL3	P'''_l	Pseudo flux due to evaporation of rain
FQSPI3	P'''_i	Pseudo flux due to deposition on snow and graupel
FQGPG3	P'''_g	Pseudo flux due to deposition on snow and graupel
FQRPR0	P_r	Flux of falling rain drops
FQIPI0	P_i	Flux of falling cloud ice
FQSPS0	P_s	Flux of falling snow
FQGPG0	P_g	Flux of falling graupel

Chapitre 6

Using DDH files : the ddhtoolbox

6.1 Purpose

ARPEGE, ALADIN and AROME models produce DDH files. The ddhtoolbox makes operations relevant to use these DDH files for scientific development and research : produce ready-to-plot profiles of variables, tendencies and fluxes (ddhi), cumulate DDH files, differentiate DDH files, make horizontal and vertical means (ddht), get the budget of prognostic variables (ddhb), etc.

6.2 Install the software

Questions : Mailto : Jean-Marcel.Piriou@meteo.fr

Untar the ddhtoolbox.tar file.

```
cd ddhtoolbox/tools
```

The install process uses the "uname -a" command to recognize the architecture of the current machine. So first type "uname -a" on the command line. Then check whether this is an already known type in the scripts install, lfa/install, and .dd2gr/src/install. If yes, and if the compiler option fits your needs, no change needs to be done. Else case, you will have to add an item in the "if ["\$os_name"] else fi" statement, or change the FC value, to set your compiler options.

1. Put the local directory in your PATH : `export PATH=. :$PATH`
2. Run install process :
 `install clean`
 `install`

6.3 Environment variables

1. Put the 3 directories in your PATH :
 - (a) **DDH tools** : ddhtoolbox/tools
 - (b) **LFA tools** : ddhtoolbox/tools/lfa
 - (c) **Plot tool** : ddhtoolbox/tools/.dd2gr/src
2. DDH tools (ddhi, ddhb) use the following environment variables, to be put in your ".profile" or ".bash_profile" files :
 `export DDHTOOLBOX= the absolute PATH of the above ddhtoolbox directory`
 `export DDHI_LIST=$DDHTOOLBOX/ddh_budget_lists/conversion_list`
 `export DDHB_BPS=$DDHTOOLBOX/ddh_budget_lists`
 `export DDH_PLOT=dd2gr`

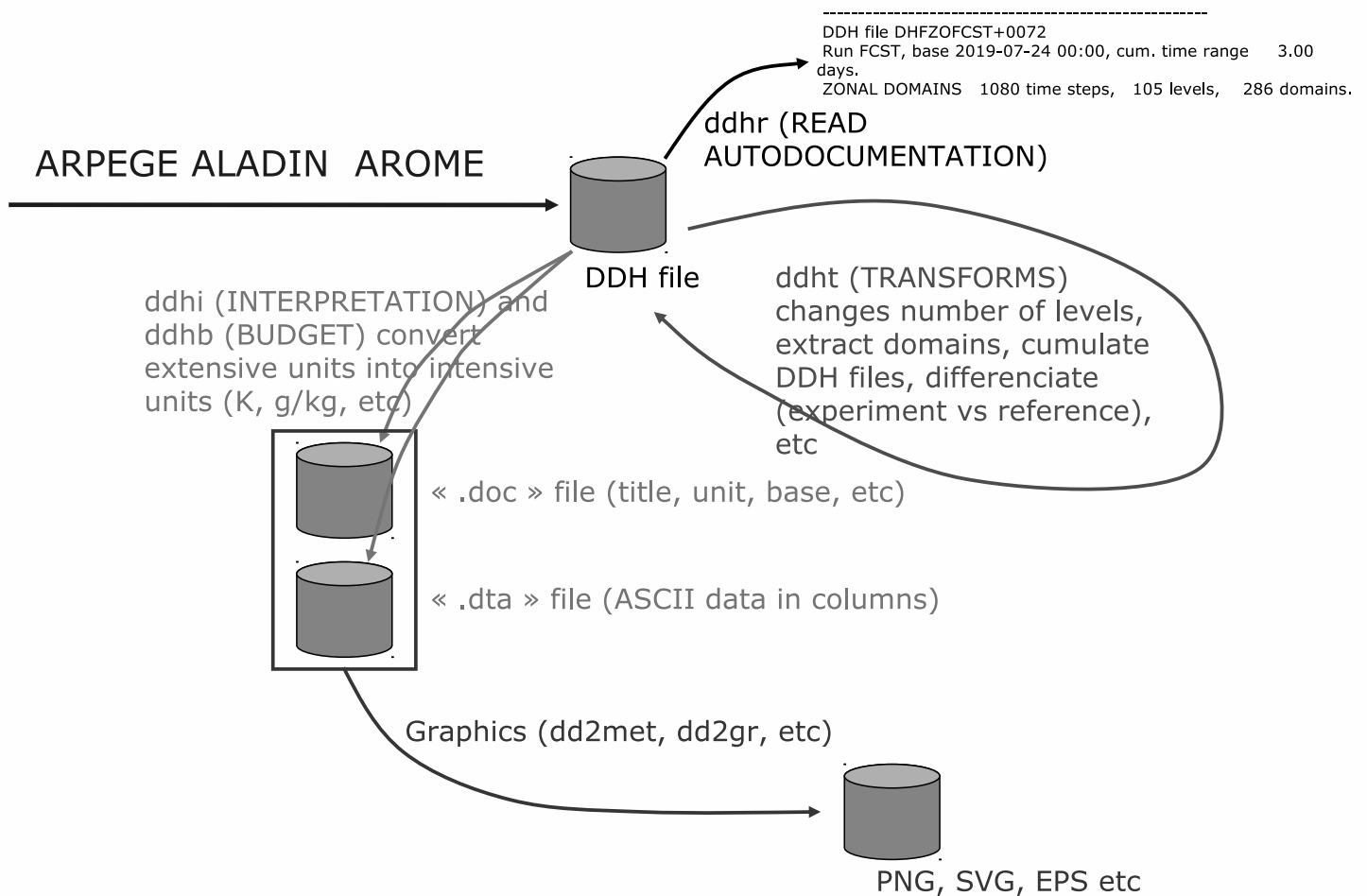
3. If the environment variable DDH_PLOT is set, some ddhtoolbox utilities call a script of your own, which makes the plot, starting from the information given by the autodocumentation file (suffix ".doc").

Graphics : all ddhtoolbox tools (ddhi, ddhr, ddhb, etc) produce ready-to-plot ASCII files. The user may then use his own graphic tool to plot these ".doc" and ".dta" files : if the environment variable DDH_PLOT is set, some ddhtoolbox tools call a script which makes the plot (creates a PNG file), starting from the information given by the autodocumentation file (suffix ".doc").

Example : the graphics may be done by dd2gr (this tool is available in the present ddhtoolbox) and visualisation by eog (or firefox), setting the environment variables in ".profile" or ".bash_profile" files : "export DDH_PLOT=dd2gr". The graphics can also be done by dd2met (written by Yves Bouteloup, based on METVIEW) setting "export DDH_PLOT=dd2met", or by any plot tool of your own.

To visualize the PNG graphics, the default tool is eog. This can be changed in editing the ddhtoolbox/tools/visu script : using the hostname function, one may choose different visualisation tools, depending on the machine (HPC, local PC, etc).

6.4 Synoptic view of ddhtoolbox utilities



ARPEGE, ALADIN or AROME models produce DDH files. Each file contains n domains, m levels, and for each domain-level all variables, fluxes and tendencies relevant to get a budget. The actions that can be performed on DDH files are described in the following sections.

6.4.1 ddhr : get autodocumentation

Get some DDH file autodocumentation on standard output : base, prediction range, etc.

Typing "ddhr" with no argument gives a documentation about the usage.

See section (6.5 p. 47) for examples of use.

6.4.2 ddht : transform

Operates transforms on DDH files : make the difference between a reference and an experiment, cumulate several DDH files, extract one or more domain(s) from a DDH file, extract one or more level(s), make an horizontal mean on all domains, make a vertical mean on all levels, etc. ddht generates in output a DDH file.

Typing "ddht" with no argument gives a documentation about the usage.

See section (6.5 p. 47) for examples of use.

Difference experiment minus reference

```
=> ddht -cDIFFE_EXP_REFE -2FEXP -1FREF -sDIFF
```

makes the difference between DDH file FEXP and FREF, the result is DIFF file. The DIFF file is a DDH file. The prediction range of FEXP and FREF have to be equal. If they differ more than 0.001%, ddht aborts.

The script ddh- makes the same operation, with a shorter command line :

```
=> ddh- FEXP FREF DIFF
```

The ddh- script calls "ddht -cDIFFE_EXP_REFE".

It may be useful, in some situations, to make the difference between 2 files having different prediction ranges : for example, to compare the mean infra-red cooling from a 24h prediction with a 6h prediction, to study spin-up effects. The script ddh- makes such a difference : if the 2 prediction ranges are different, ddh- modifies the prediction range from one file, modifies all fluxes and tendencies accordingly (done through the ddhmech script), and then makes the "ddht -cDIFFE_EXP_REFE" difference.

See section (6.5 p. 47) for examples of use.

6.4.3 ddhi : interpretation

ddhi makes an interpretation of the raw data from a DDH file, in order to get ready to plot data, with intensive units (K/day, g/kg, etc).

Typing "ddhi" with no argument gives a documentation about the usage.

Example : starting from a DDH file "DHFDLFCST+0024.domaine4", one needs to get an ASCII file containing the profile of water vapour q_v and temperature T . Create an ASCII file containing the list of articles :

```
lxgmap2:/home/piriou/ftn/ddh/ddhtoolbox/ddh_files/arome/cy35t1_arome_france_c744 => cat mylist
VQV1
VCT1
```

then type "ddhi DHFDLFCST+0024.domaine4 -lmylist" :

```

lxgmap2:/home/piriou/ftn/ddh/ddhtoolbox/ddh_files/arome/cy35t1_arome_france_c744 => ddhi DHFDLFCST+0024.domaine4 -lmylist
default list file:
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/conversion_list
-----
---DDHI-CHAMPS-----
Fichier d'entree: DHFDLFCST+0024.domaine4
calling lisc
lisc:/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/conversion_list
lisc: read          413  fields.
-----
---DDHI-COORDONNEES---
-----
DHFDLFCST+0024.domaine4.tmp.VCT1.dta
DHFDLFCST+0024.domaine4.tmp.VCT1.doc
DHFDLFCST+0024.domaine4.tmp.VQV1.dta
DHFDLFCST+0024.domaine4.tmp.VQV1.doc

```

ddhi reads the DDH file, reads the VCT1 and VQV1 articles, converts the units of these data : for example the VCT1 article is $c_p T \frac{\delta p}{g}$, ddhi converts it into T , and thus divides by c_p and by the $\frac{\delta p}{g}$ field. The conversion uses a conversion list file (whose name is given to ddhi by the DDHI_LIST environment variable), which tells ddhi how to convert each DDH article.

ddhi produces in output ASCII files :

- a ".doc" file per field, containing autodocumentation (title, unit, base, prediction range, etc)
- a ".dta" file per field, containing the data in columns.

To know the complete list of variables, tendencies and fluxes that can be transformed into ASCII data by ddhi, type "lfaminm FDDH", where FDDH is the name of the DDH file. This gives the list of all articles. The variables are article names beginning with "V", the tendencies are article names beginning with "T", the fluxes are article names beginning with "F". These article names can be put in the "mylist" file as described above. If one of these articles is not present in the conversion list file DDHI_LIST, ddhi will not know how to convert it. In this case, simply edit the DDHI_LIST file, and add a line describing how this article has to be converted.

See section (6.5 p. 47) for examples of use.

6.4.4 ddhb : budgets of prognostic variables

ddhb is a tool to make the budget of prognostic variables, starting from a DDH file. Typing "ddhb" with no argument gives a documentation about the ddhb use.

This DDH file should contain only one domain. If it is not the case, use ddht to extract a single domain from your multi-domain DDH file.

See section (6.5 p. 47) for examples of use.

Get a first budget plot

Example of ddhb use :

```
"ddhb -v QV -i DHFDLALAD+0036".
```

In this example, one asks for the budget of the prognostic variable is QV (water vapour), from the file DHFDLALAD+0036. What ddhb basically does, as one types -v QV, is to read in the DDH file the list of articles containing fluxes or tendencies of QV : all articles "FQV*" and "TQV*" of the DDH file are used to build the QV budget.

The ddhb script then writes ASCII ready to plot files ; two kinds of files are produced :

1. **Data** files (suffix : dta) : in ASCII columns.
2. **Autodocumentation** files (suffix : doc) : title, unit, date of the prediction run, etc.

See section (6.5 p. 47) for examples of use.

More advanced use to get budget plots

The user may also create his own directive files. For example, to change the legends of the budgets (and translate them to French, German, etc), or to customize the scientific budgets (change the list of file articles to be used for a given physical process). In this case, two methods :

1. **Create your own directive file, "from scratch" :** "ddh2fbl FDDH DIR" will read the FDDH file, and produce the directive files on the \$DDHB_BPS/DIR directory. The acronym "FBL" stands for Field Budget List. Then, one simply needs to type

"ddhb -v DIR/VAR -i FDDH" to get the budget of the variable VAR. If one wants to modify the legends of the curves, one edits the \$DDHB_BPS/DIR/VAR.fbl file, and then reruns ddhb.

How does ddh2fbl work ? It reads inside the DDH file all article names, lists the articles of the type V ??0 (examples : ??=CT, QV, etc). For each of these variables, lists all F ??* and T ??* articles. And writes the directive file containing this list. The resulting list is thus consistent with the DDH file. ddh2fbl makes the assumption that all budget items are articles beginning with F or T. This rule, presently true in ARPEGE - ALADIN - AROME, should thus be respected in the future to guarantee a proper work of ddh2fbl.

2. **Modify existing reference directive files :**

```
find $DDHB_BPS -name "*.fbl" -print
```

to see the complete list of physics or budget packages available for use in ddhb, and which variables. Copy a directory containing directive files under your own name, and then modify it. Example :

```
lxgmap2: => find $DDHB_BPS -name "*.fbl" -print
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arome_cy35t1/QG.fbl
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arome_cy35t1/QR.fbl
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arome_cy35t1/QL.fbl
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arome_cy35t1/QI.fbl
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arome_cy35t1/QS.fbl
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arome_cy35t1/CT.fbl
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arome_cy35t1/QV.fbl
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arome_cy35t1/TE.fbl
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arpege/KK.fbl
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arpege/QT_old.fbl
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arpege/QT.fbl
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arpege/CT_simplified.fbl
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arpege/QV_2006-06_and_previous.fbl
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arpege/QV_simplified.fbl
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arpege/CT.fbl
/home/piriou/ftn/ddh/ddhtoolbox/ddh_budget_lists/arpege/QV.fbl
lxgmap2: => cp -r $DDHB_BPS/arpege/ $DDHB_BPS/myown_arpege
lxgmap2: => vi $DDHB_BPS/myown_arpege/CT.fbl
lxgmap2: => ddhb -v myown_arpege/CT -i DHFDLALAD+0036
```

6.5 Using the ddhtoolbox, step by step

6.5.1 Vertical profile of variables, tendencies or fluxes

```
cd ddhtoolbox/demonstration/arome_hovmoller
cd ddhpack_dlimited
```

In this directory, the DDH files are DHFDLFCST+????, from 1h prediction range to 36h. Each DDH file contains several domains. These domains were specified in the NAMDDH namelist from the ARPEGE - ALADIN - AROME which produced these DDH files.

To get the autodocumentation of a DDH file :

```
ddhr DHFDLFCST+0032
```

This indicates prediction time, number of time steps, date of analysis of this prediction, number of levels of the model, and the number of domains inside this DDH file.

"ddhr" with no argument gives a synopsis of this tool.

```
ddhr -ddd DHFDLFCST+0032
```

gives more detail about each domain inside this DDH file : latitude and longitude of single points, corners for rectangular domains, etc.

If one is interested in the 11rd domain, for example, let us first extract this domain from the DDH file :

```
ddht -cEXTRAIT_DOMAIN -1DHFDLFCST+0032 -E11 -sDHFDLFCST+0032.mydomain
```

Creates the output file DHFDLFCST+0032.mydomain. This file contains only one domain, the 11rd domain from DHFDLFCST+0032.

"ddht" with no argument gives a synopsis of this tool.

To check the content of this new file :

```
ddhr DHFDLFCST+0032.mydomain
```

Indicates

```
-----
DDH file DHFDLFCST+0032.mydomain
Run G6ZJ, base 2023-10-06 00:00, cum. time range 32.00 h.
LIMITED AREA DOMAINS 2304 time steps, 90 levels, 1 domains.
```

This DDH file contains a single domain. To get more information about this domain, type as previously "ddhr -ddd" :

```
ddhr -ddd DHFDLFCST+0032.mydomain
```

Indicates this domain is a point, longitude= 0.3835, latitude= 43.1225.

Which fields are in this single domain DDH file, and thus can be plotted? To get a list :

```
lfaminm DHFDLFCST+0032.mydomain
```

Prompts the list of all articles of this file. Such a list can be got because the DDH files are autodocumented binary files, written in the LFA format. To get more information about the LFA format, see the documentation ddhtoolbox/tools/lfa/documentation/lfa_english.pdf.

The list, output from lfaminm above, contains many articles names, like :

```
l= 90, min= 0.3547E+07 max= 0.6419E+08 mea= 0.2940E+08 rms= 0.3108E+08 |R4| VCT1
l= 90, min= 0.3531E+07 max= 0.6474E+08 mea= 0.2944E+08 rms= 0.3113E+08 |R4| VCT0
l= 90, min= 0.1032 max= 78.36 mea= 27.97 rms= 34.10 |R4| VHR1
l= 90, min= 0.1135 max= 146.5 mea= 45.38 rms= 62.15 |R4| VHR0
l= 91, min= 0.1439E+08 max= 0.2007E+08 mea= 0.1685E+08 rms= 0.1693E+08 |R4| FCTRAYS0
l= 91, min= -0.3201E+08 max= -0.9774E+07 mea= -0.2092E+08 rms= 0.2211E+08 |R4| FCTRAYTH
l= 90, min= -0.2081 max= 0.000 mea= -0.3209E-02 rms= 0.2353E-01 |R4| TCTIMLT
l= 90, min= 0.000 max= 0.5823E-06 mea= 0.8982E-08 rms= 0.6584E-07 |R4| TQLIMLT
l= 90, min= -0.5823E-06 max= 0.000 mea= -0.8982E-08 rms= 0.6584E-07 |R4| TQIIMLT
```

Article names beginning with "V" are variables, the second and third characters are the variable itself (CT for $C_p T$, QV for water vapour, UU for zonal wind, VV for meridional wind, KK for kinetic energy, etc). "T" are cumulated tendencies, "F" cumulated fluxes. For example above, FCTRAYS0 is the cumulated flux of solar radiation, FCTRAYTH the cumulated thermal radiation flux, VCT0 is initial temperature ($CT = C_p T$), VCT1 final temperature, VHR0 initial relative humidity, VHR1 final relative humidity, etc.

To get these articles in ready to plot ASCII files, first write a list of desired articles in an ASCII file :

```
vim listv
```

In listv file one writes for example the lines :

```
VCT1
VCT0
VHR1
FCTRAYS0
```

The ddhi tool is used to read these articles and convert the data into a ready to plot information :

```
ddhi -l1listv DHFDLFCST+0032.mydomain
```

This writes out the following files :


```
DHFDLFCST+0032.mydomain.tmp.VCT0.dta
DHFDLFCST+0032.mydomain.tmp.VCT0.doc
DHFDLFCST+0032.mydomain.tmp.VCT1.dta
DHFDLFCST+0032.mydomain.tmp.VCT1.doc
DHFDLFCST+0032.mydomain.tmp.VHR1.dta
DHFDLFCST+0032.mydomain.tmp.VHR1.doc
DHFDLFCST+0032.mydomain.tmp.FCTRAYS0.dta
DHFDLFCST+0032.mydomain.tmp.FCTRAYS0.doc
```

2 files per field : the ".doc" file is an autodocumentation, to be read by your graphic tool : title, unit, date, name of the field, etc. The ".dta" is an ASCII file with columns. The last column is the field, the firsts columns are the coordinates.

To plot this filed with the graphic tool dd2png :

```
dd2png DHFDLFCST+0032.mydomain.tmp.VCT1.doc DHFDLFCST+0032.mydomain.tmp.VCT1.png
```

dd2png reads the ".doc" file and plots a PNG graphic in the file name given by the second argument. This plots the vertical profile of final temperature (at 32h prediction time), of this 11rd domain. An example of the resulting plotted profile of temperature is given in figure (6.1 p. 49).

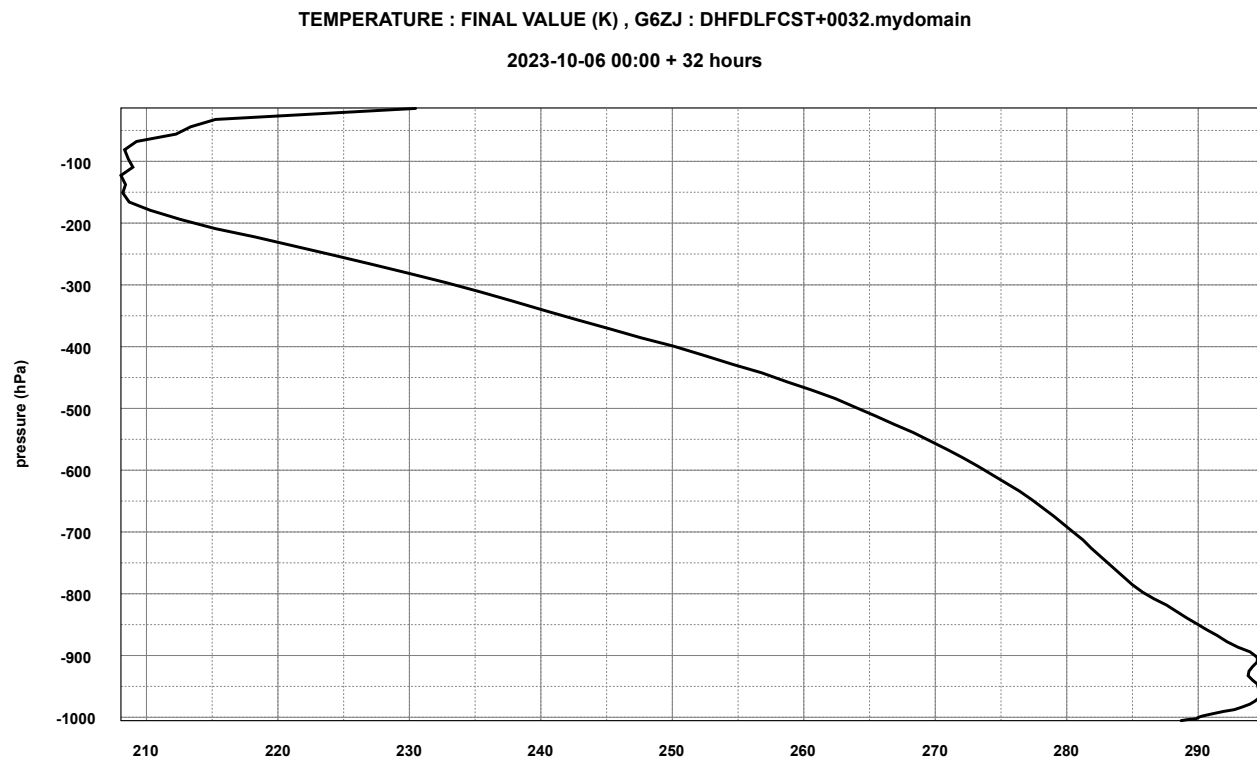


FIGURE 6.1 – Vertical profile of temperature at 32h prediction time.

ddhi converts the data from the DDH file into ready to plot data. For example, temperature is written the in the file as $c_p T \delta p / g$. ddhi divides by c_p , normalizes by the field $\delta p / g$ (which is in the VPP0 file article). All these actions can be parameterized, they are written in the file given by the unix environment variable DDHI_LIST. Example, in your .bash_profile you may have

```
export DDHTOOLBOX=/home/piriou/ftn/ddh/ddhtoolbox
export DDHI_LIST=$DDHTOOLBOX/ddh_budget_lists/conversion_list
export DDHB_BPS=$DDHTOOLBOX/ddh_budget_lists
```

If one gets a warning message "DDHI/WARNING : this article is not in the above conversion list lisc", this means that the article written in the list entered as "-l" to ddhi is missing from the conversion_list.

To add it (or customize or translate names), one can edit this `conversion_list` :

```
vim $DDHI_LIST
```

For example, the line for VCT1 (final temperature) is

```
VCT1      VCT1      TEMPERATURE : FINAL VALUE      K      0.0009953132115872  0
```

The constant 0.0009953132115872 is $1/c_p$. The clear name if the field can be customized (translation, for example). The 4 first characters indicate the conversions to be done on the field before writing it as an ASCII column ready to plot :

1. The 1st character is not used by ddhi.
2. The 2nd character indicates if operations are to be done on 2 fields : I (none), S (c_1+c_2), D (c_2-c_1), F ($((c_2-c_1) / \text{prediction range})$), etc.
3. The 3rd character is vertical processing, can be I (none), C (vertical processing : cumulate), T (vertical processing : differentiate $j_{lev} - (j_{lev}-1)$).
4. The 4th character is normalization, can be I (no normalization), 0 (normalization by initial pressure), 1 (normalization by final pressure), P (normalization by time cumulated pressure), M (normalization by time cumulated pressure divided by prediction range), etc.

6.5.2 Plotting tephigrams from DDH files

For physical interpretation, tephigrams are useful plots. To get a tephigram from any domain from a DDH file, follow this example :

```
cd ddhtoolbox/demonstration/arome_tephigram
```

In this directory the shell script uses `ddht` to extract from a DDH file the point closest to Toulouse, given by its longitude and latitude :

```
ddht -cEXTRAIT_DOMAIN -1$fddh -s$fddh.meteopole.lfa -E+001.374_+43.575_Toulouse-Meteopole
```

The argument `-E` specifies the longitude and latitude. This argument could be given as

```
-E+001.374_+43.575
```

this leads to the same result as giving

```
-E+001.374_+43.575_Toulouse-Meteopole
```

All what is written after the third `"_"` is like a comment, it is ignored by `ddht`, this comment is useful simply to make the reading of scripts more friendly, with a clear indication of the location of given (longitude, latitude) coordinates.

The script then converts the DDH file into a MUSC single column file type, using `ddh2scm`. This file can be read then by the tool `ms` (for Model Sounding), which computes vertical profiles of θ , θ_w , $\text{iso-}\theta'_w$, etc. The tool `ms` also creates an autodocumentation text file (suffixed `".doc"`), ready to be used by any plot tool.

To make the plot, the script calls `dd2gr`, this reads the `".doc"` file, makes the plot in SVG, and then converts this SVG into PNG :

```
#
#-----
# Plot.
#-----
dd2gr $fddh.meteopole.lfa.Dom001.Var_fin.scm.ms.tmp.theta.doc $fddh.meteopole.tephigram.svg
convert $fddh.meteopole.tephigram.svg $fddh.meteopole.tephigram.png
```

The PNG file can now be visualized, shown in figure (6.2 p. 51).

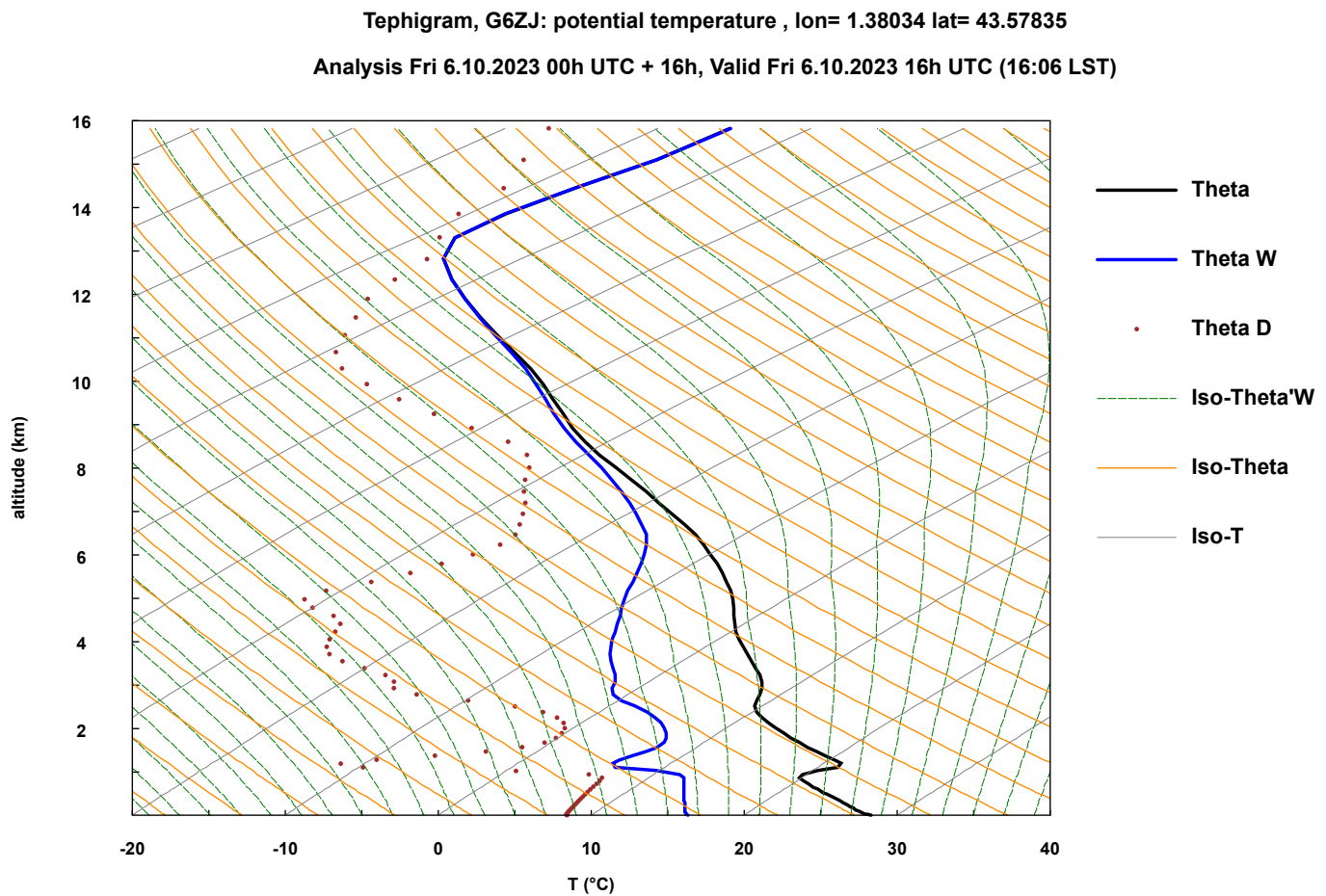


FIGURE 6.2 – Tephigram plot of a DDH domain, plotted starting from any DDH file or domain, and then using ddht to extract, ddh2scm to convert into a SCM MUSC file, ms to create the tephigram sounding, dd2gr to plot the tephigram.

6.5.3 Fluxes, with / without conversion to tendencies

```
cd ddhtoolbox/demonstration/arpege_budget/ddhpack_dlimited
```

DDH files are available from 1h to 78h prediction range. Lets study the turbulence flux of enthalpy, whose article in the DDH file is FCTTUR. The whole list of articles can be got for example with

```
lfaminm DHFDLFCST+0017
```

Each DDH file contains the fluxes and tendencies cumulated from step 0 to current time of integration. For example, DHFDLFCST+0012 contains 12h integration fluxes and tendencies, DHFDLFCST+0077 contains 77h.

The script "ddh_decumulifie" decumulates :

```
ddh_decumulifie DHFDLFCST+????
```

will loop over all DDH files of the current directory, and make the difference between each file and the previous. This results in creating the DHFDLFCST+????.instantane files.

One can check the results using ddhr :

```
ddhr DHFDLFCST+0016
```

indicates a 16h prediction time.

```
ddhr DHFDLFCST+0017
```

indicates a 17h prediction time.

```
ddhr DHFDLFCST+DHFDLFCST+0017.instantane
```

indicates a 1h prediction time, relevant for the 16h to 17h prediction period.

We now extract from this file the point closest to Toulouse :

```
ddht -cEXTRAIT_DOMAIN -1DHFDLFCST+0017.instantane -sDHFDLFCST+0017.instantane.meteopole.lfa -E+001.374+43.575
```

To get the profile of turbulence, one writes with an editor a list file (lets call it listv) with this article :

```
cat listv
FCTTUR
```

To get data in ASCII colums from this turbulent process, we use ddhi :

```
ddhi -l1listv -1VZ DHFDLFCST+0017.instantane.meteopole.lfa
```

The -1VZ arguments asks for Vertical Z coordinate as 1st ASCII column in the output file (other options : VP (pressure), VN (levels), by default : pressure). This creates the following files :

```
DHFDLFCST+0017.instantane.meteopole.lfa.tmp.FCTTUR.dta
DHFDLFCST+0017.instantane.meteopole.lfa.tmp.FCTTUR.doc
```

To get a plot :

```
dd2png DHFDLFCST+0017.instantane.meteopole.lfa.tmp.FCTTUR.doc DHFDLFCST+0017.instantane.meteopole.lfa.tmp.FCTTUR.png
```

The PNG file contains the plot of the vertical profile of T tendency (K/day) due to turbulence, as by default ddhi converts fluxes into tendencies.

To get raw fluxes, use the option -cNON (no conversion) :

```
ddhi -l1listv -1VZ -cNON DHFDLFCST+0017.instantane.meteopole.lfa
```

The flux is not converted into tendency. The plot

```
dd2png DHFDLFCST+0017.instantane.meteopole.lfa.tmp.FCTTUR.doc DHFDLFCST+0017.instantane.meteopole.lfa.tmp.FCTTUR.png
```

now shows a turbulence flux, in J/m2.

As we are interested to zoom in the PBL, ddhi can avoid producing the data over 1 km :

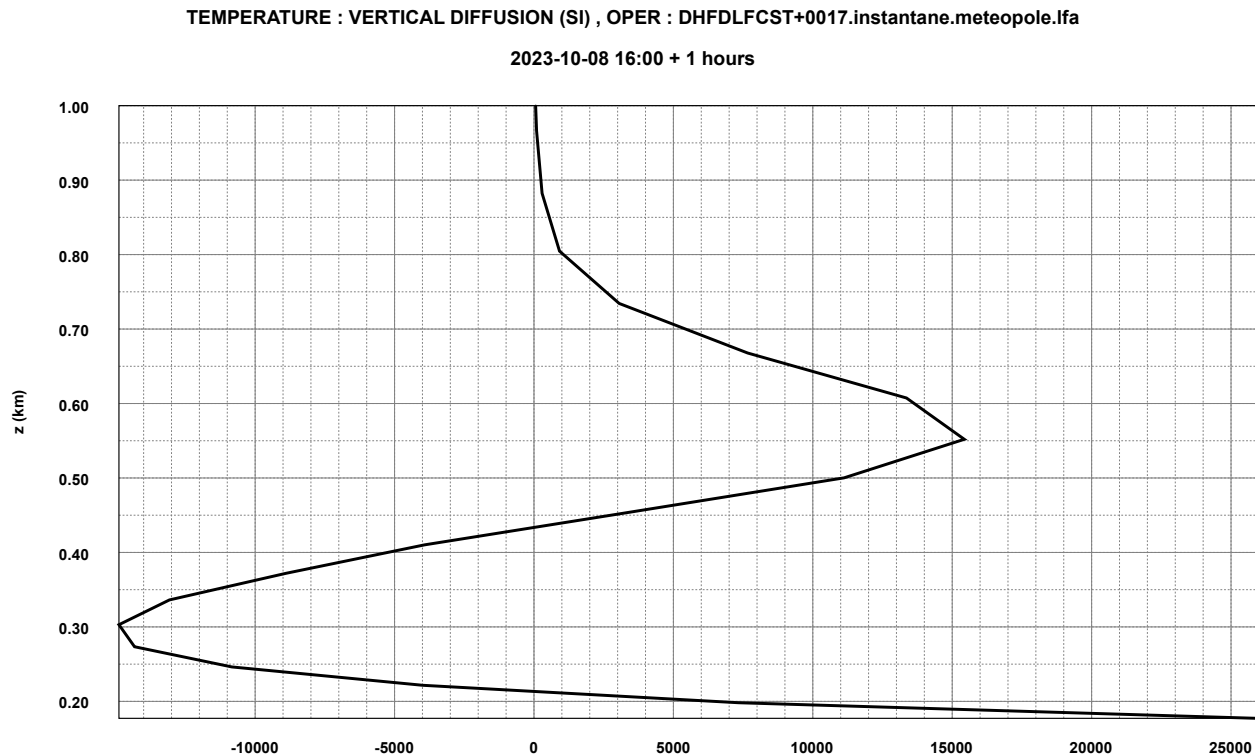


FIGURE 6.3 – Vertical profile of turbulence heat flux, over Toulouse, between 16h and 17h prediction times, in $J \cdot m^{-2}$.

```
ddhi -l1listv -1VZ -cNON DHFDLCST+0017.instantane.meteopole.lfa -ymax1.
```

Redo the plot :

```
dd2png DHFDLCST+0017.instantane.meteopole.lfa.tmp.FCTTUR.doc DHFDLCST+0017.instantane.meteopole.lfa.tmp.FCTTUR.png
```

now shows a turbulence flux, in J/m^2 , below 1 km height. This plot is shown in figure (6.3 p. 53).

Exercise : PBL height may be defined as the level at which the turbulence flux value is 10% of that at surface. Using output files from ddhi and scripts of your own, plot the temporal evolution of PBL height from 1 to 78h. An example of resulting plot of this time evolution of PBL height is shown in figure (6.4 p. 54).

Compare, for some prediction ranges, this PBL height to that seen on a tephigram. A tephigram can be created at any prediction range :

```
ddh2scm DHFDLCST+0017.instantane.meteopole.lfa
```

converts the DDH file into a single-column MUSC-type file. The file created is

```
DHFDLCST+0017.instantane.meteopole.lfa.Dom001.Var_fin.scm
```

To compute a tephigram from this vertical profile :

```
ms DHFDLCST+0017.instantane.meteopole.lfa.Dom001.Var_fin.scm
```

ms stands for "model sounding". The following files have been created :

```
DHFDLCST+0017.instantane.meteopole.lfa.Dom001.Var_fin.scm.ms.tmp.theta.doc
```

and several ASCII dta files ".dta". To plot the sounding :

```
dd2png DHFDLCST+0017.instantane.meteopole.lfa.Dom001.Var_fin.scm.ms.tmp.theta.doc DHFDLCST+0017.instantane.meteopole.lfa.Dom001.Var_fin.scm.ms.tmp.theta.png
```

This plot is shown in figure (6.5 p. 54). The PBL height can be seen on this tephigram. So, are the PBL heights from "10% of turbulence flux" consistent to those seen from such tephigrams?

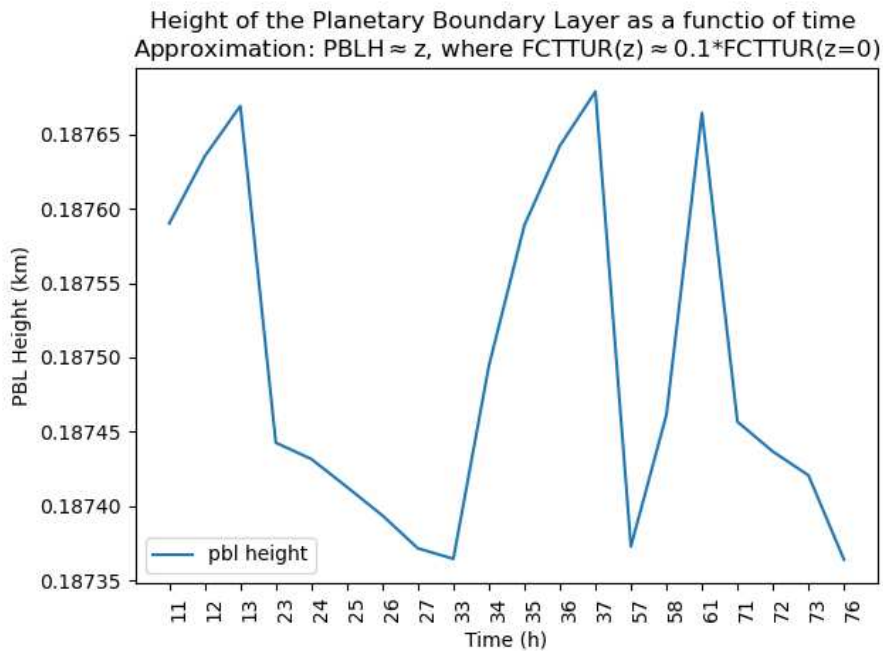


FIGURE 6.4 – Example of PBL height evolution, based on the "10% of surface turbulence flux", and the above mentioned DDH files (source Panu Maalampi, Oct 2023).

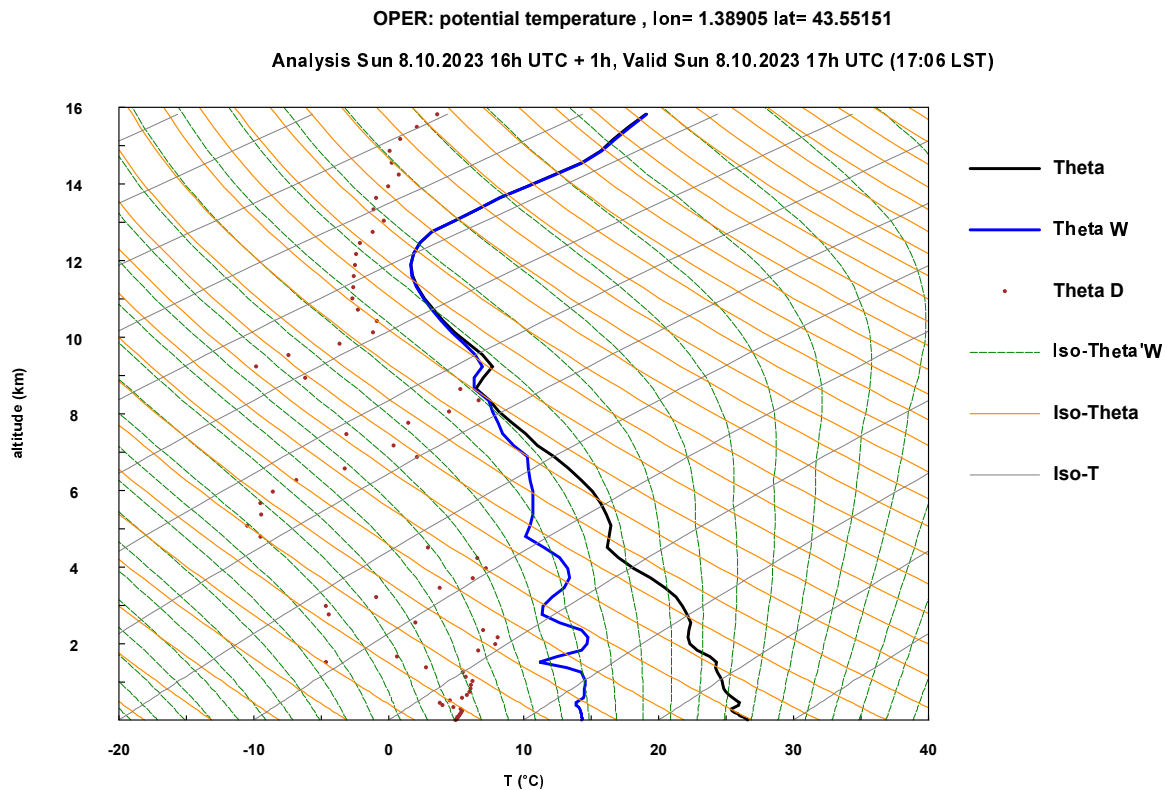


FIGURE 6.5 – Tephigram over Toulouse at 17 UTC.

6.5.4 Plotting ARPEGE/AROME budgets : example of studying infra-red / turbulence interaction

This section produces temperature budgets at different time and space scales, to study the interaction between infra-red and turbulence processes.

The ARPEGE operational model produces zonal DDH files. Any DDH file contains one or several domains. A zonal DDH file is a file containing as many domains as zonal bands.

```
cd ddhtoolbox/demonstration/arpege_interaction_rad_vs_turbulence/data_ddh_monthly
```

The DDH files in this directory are dhfzoprod+0072.r00.lfa.*.lfa. The zonal files were produced every day in ARPEGE operational predictions, and then cumulated in time to get monthly mean zonal DDH files. How to cumulate DDH files will be presented later in this documentation.

```
ddhr dhfzoprod+0072.r00.lfa.20160201.lfa
```

gives autodocumentation information about a given DDH file. We get the output :

```
ddhr dhfzoprod+0072.r00.lfa.20160201.lfa
-----
DDH file dhfzoprod+0072.r00.lfa.20160201.lfa
Run arpege_europe_oper, base 2016-02-01 00:00, cum. time range    87.00 days, time range for var.    3
ZONAL DOMAINS  20880 time steps,   105 levels,       30 domains.
```

This zonal DDH file contains 30 zonal bands (30 domains). Its cumulated time range is 87 days, as it was cumulated from 29 predictions (29 days in February 2016) at 3 days prediction range. The model producing these zonal DDH file has 105 levels.

In this example-section, we are interested in studying the evolution, among years of ARPEGE predictions, of the interaction between infra-red radiation and turbulence. To get a budget of a DDH file, we need first to have a DDH file containing a single domain. Here we will make this study in global mean. To produce a global mean from a zonal DDH file, we use the ddht tool, which reads a DDH file with n domains, and generates another DDH file, containing a single domain, the mean one :

```
ddht -cMOY_HORIZ -ldhfzoprod+0072.r00.lfa.20160201.lfa -sdhfzoprod+0072.r00.lfa.20160201.lfa.mh
```

-cMOY_HORIZ means "horizontal mean", -l is the input file, -s the output file. To check we now have a single domain in this output file :

```
ddhr dhfzoprod+0072.r00.lfa.20160201.lfa.mh
```

now gives

```
ddhr dhfzoprod+0072.r00.lfa.20160201.lfa.mh
-----
DDH file dhfzoprod+0072.r00.lfa.20160201.lfa.mh
Run arpege_europe_oper, base 2016-02-01 00:00, cum. time range    87.00 days, time range for var.    3
GLOBAL DOMAIN  20880 time steps,   105 levels,       1 domains.
```

It contains a single horizontal domain, the whole Earth.

To make a temperature budget the command is

```
ddhb -v oper/CT -i dhfzoprod+0072.r00.lfa.20160201.lfa.mh -o dhfzoprod+0072.r00.lfa.20160201.lfa.mh.svg
```

The argument -v oper/CT means we want to make the budget of CT (temperature, $c_p T$), using the budget list directives written in file ddhtoolbox/ddh_budget_lists/oper/CT.fbl. These budget directives are text files, that can be modified, to change the language (create files in English, French, etc), the units, or update to new code cycles (new ways of dealing physical parameterizations).

In output, ddhb writes a graphic file (-o option). This SVG file can be converted into PNG one for example using convert :

```
convert dhfzoprod+0072.r00.lfa.20160201.lfa.mh.svg dhfzoprod+0072.r00.lfa.20160201.lfa.mh.png
```

Exercise : redo such a budget for all February files available. How does the interaction between infra-red T tendency and turbulence T tendency evolve among the years? What may be the explanation for infra-red T tendency becoming positive in the recent period of time?

To look at this at much smaller scale and smaller time range, such a budget can be done in an AROME prediction :

```
cd ddhtoolbox/demonstration/arome_budget/ddhpack_dlimited
```

The DDH files here are DHFDLFCST+???? . To get the budget for example between 12h and 13h prediction time, lets make the difference in time between the file at 13h and that at 12h :

```
ddht -cDIFFE_EC2_EC1 -2DHFDLFCST+0013 -1DHFDLFCST+0012 -sDHFDLFCST+0012-0013
```

-cDIFFE_EC2_EC1 indicates a difference between 2 DDH files from the same prediction, at different prediction times. -1 and -2 arguments are the input files, -s is the output file receiving the difference.

```
ddhr DHFDLFCST+0012-0013
```

Indicates

```
-----
DDH file DHFDLFCST+0012-0013
Run G6ZJ, base 2023-10-06 12:00, cum. time range      1.00 h.
LIMITED AREA DOMAINS      72 time steps,      90 levels,      39 domains.
```

The prediction is now 1h : from 12h to 13h. This file contains 39 domains. Lets extract the domain closest to Toulouse :

```
ddht -cEXTRAIT_DOMAIN -E+001.374_+43.575_Toulouse-Meteopole -1DHFDLFCST+0012-0013 -sDHFDLFCST+0012-0013.meteopole.lfa
```

Lets plot the temperature budget between 12h and 13h.

```
ddhb -v AROME_G6ZJ_fbl/CT -i DHFDLFCST+0012-0013.meteopole.lfa -o DHFDLFCST+0012-0013.meteopole.svg
convert DHFDLFCST+0012-0013.meteopole.svg DHFDLFCST+0012-0013.meteopole.png
```

Looking at this picture -see figure (6.6 p. 56)-, showing a temperature budget over Toulouse during a 1h time period, we see the infra-red tendency close to surface having much larger values, compensated by a large negative value from turbulence. This means that a large part of CLS heating by infra-red radiation is now explicitly done by the radiation scheme. In this case, why does turbulence still parameterize net sensible heating at surface through the difference ($T_s - T_n$), which in principle mimics net infra-red radiation?

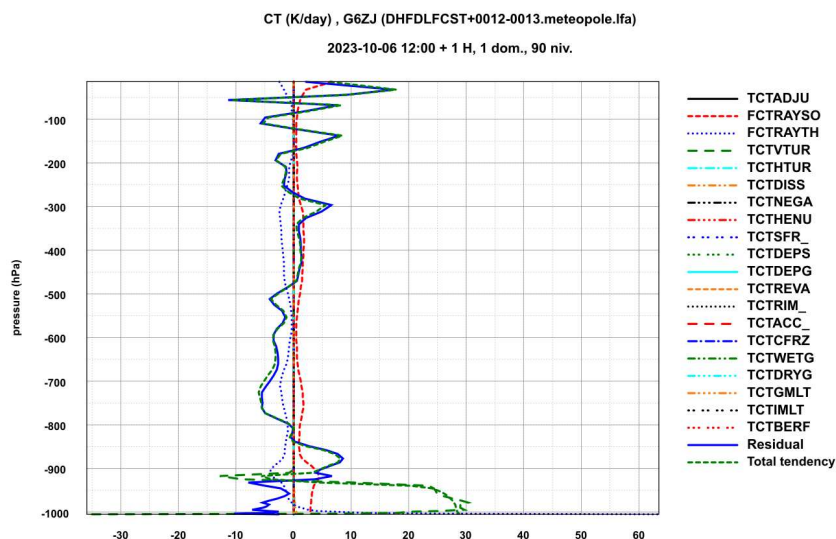


FIGURE 6.6 – Temperature budget predicted by AROME G6ZJ over Toulouse, between 12 and 13 UTC, 6 October 2023.

6.5.5 Plotting ARPEGE/AROME budgets : customizing the budget names and categories

```
cd ddhtoolbox/demonstration/customize_budgets
```

The script in this directory computes the DDH file relevant for the evolution between 9 and 10h prediction times, using `ddht -cDIFFE_EC2_EC1`. Then it extracts from this resulting 1h prediction time DDH file the point closest to Toulouse, using `ddht -cEXTRAIT_DOMAIN`.

Using this DDH file containing a single domain over Toulouse, it makes the temperature (CT) budget with the `ddhb` tool, using first a standard budget directive file (in the `AROME_G6ZJ_fbl` directory). This gives the plot left hand side (LHS) of the figure (6.7 p. 57).

On the plot right hand side (RHS) of the figure (6.7 p. 57), the customized title "Rad Solar + Thermal" refers to a field that is the sum of the 2 ones "FCTRAYS0 and FCTRAYER" on the left. Samely, the title "Turb vert + horiz" refers to the sum of the 2 ones "TCTVTUR + TCTHTUR".

How can such customized budget be generated ? To get the budget shown on the RHS, the script uses another budget directive file, customized, available in the `AROME_customized_fbl` directory.

Comparing (with a text editor) the `AROME_G6ZJ_fbl/CT.fbl` and `AROME_customized_fbl/CT.fbl` files, one can see the syntax of such directive files, and how to modify them to get customized budgets.

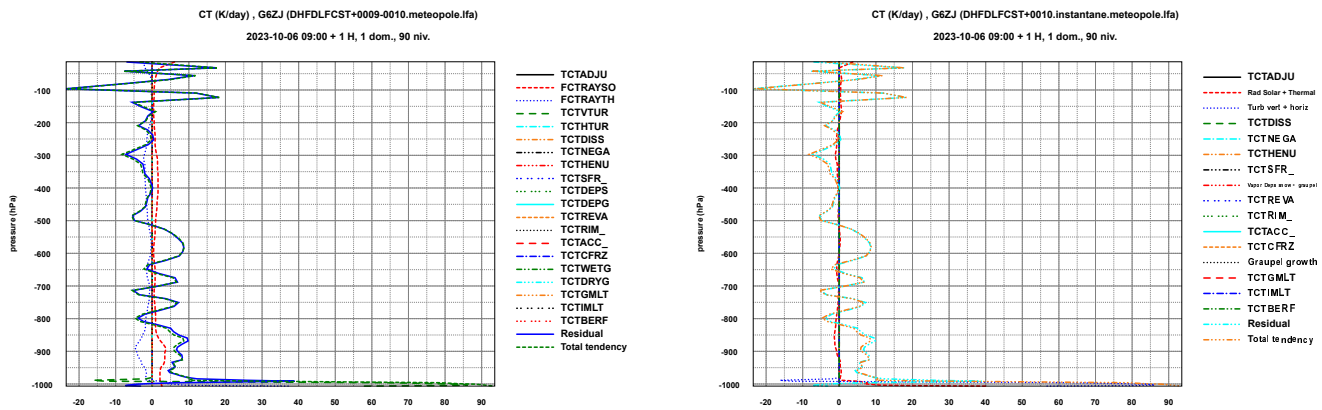


FIGURE 6.7 – Temperature budget predicted by AROME G6ZJ over Toulouse, between 0 and 10 UTC, 6 October 2023. Source Panu Maalampi, October 2023.

6.5.6 Hovmöller diagrams of the difference between an experiment and a reference

To compare an experiment versus a reference, like validating a new version of 3D code, or to check the impact of modification in physics or dynamics, Hovmöller diagrams of differences are useful to check whether differences appear from first time step (difference in analysis), or gradually (differences in physics / dynamics), or are larger in the firsts time steps (spin-up due to imbalance between analysis and model equations).

```
cd ddhtoolbox/demonstration/compare_2_predictions_hovmoller
```

In this directory the script uses `ddht` (extract a domain over Toulouse), `ddht` (to interpolate a vertical grid on another, as the 2 DDH files were produced by models with a different number of levels, before computing the difference), `ddhi` (to produce vertical profiles ready to plot) and `dd2gr` to make the plot. The results can be seen on figures (6.8 p. 58) (graphic tool `dd2gr`) and (6.9 p. 59) (graphic tool `matplotlib`).

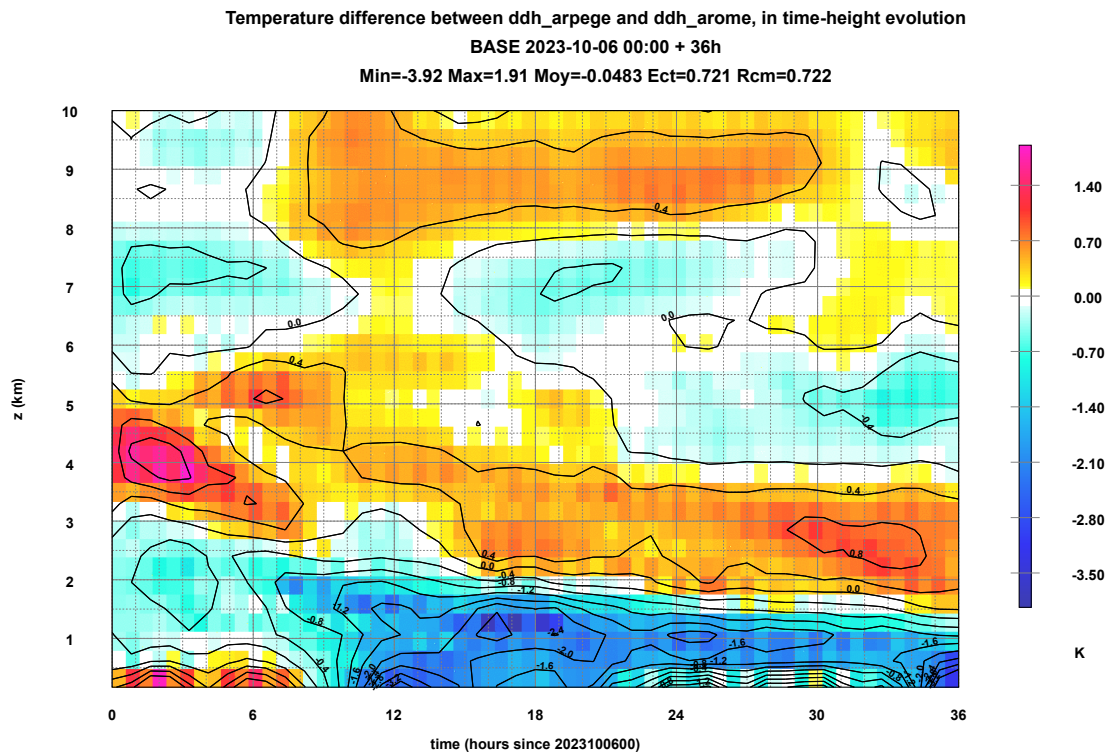


FIGURE 6.8 – Difference in temperature between an AROME (G6ZJ) prediction and an ARPEGE (OPER) one, over Toulouse, in time-height coordinates (Hovmöller diagram). Plotted with the dd2gr tool (Jean-Marcel Piriou).

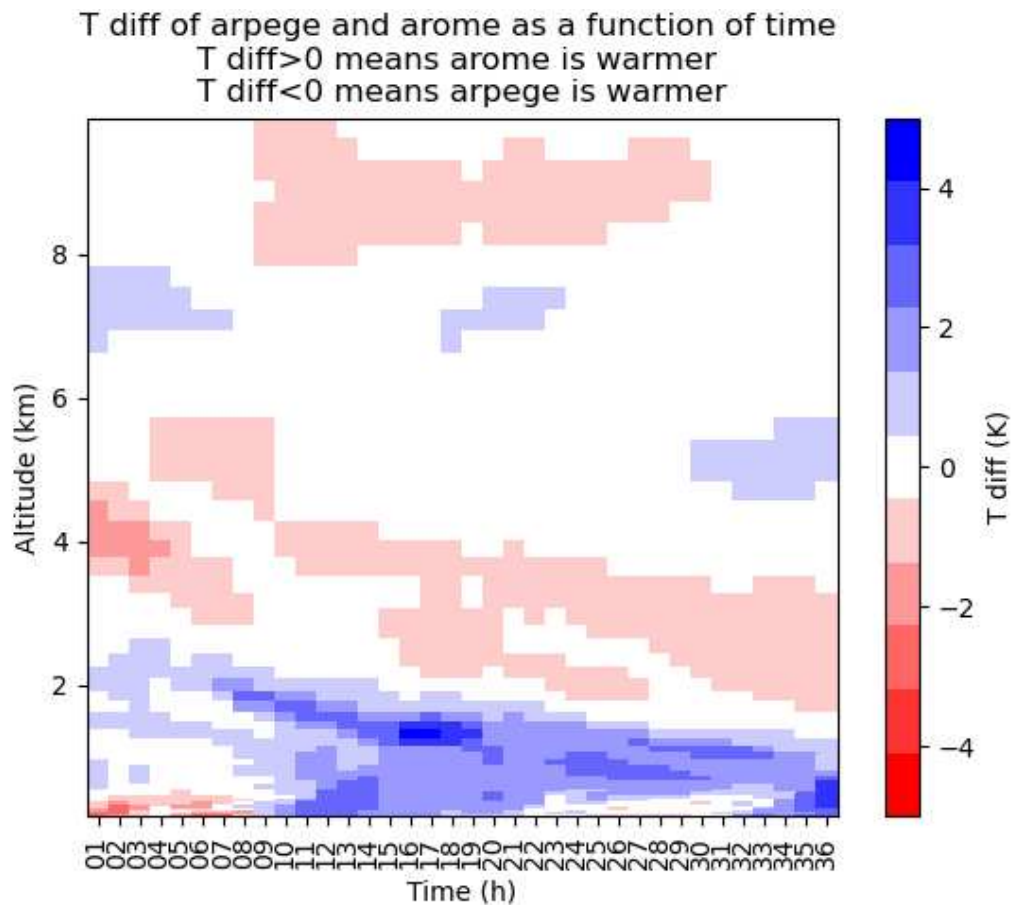


FIGURE 6.9 – Difference in temperature between an AROME (G6ZJ) prediction and an ARPEGE (OPER) one, over Toulouse, in time-height coordinates (Hovmöller diagram). Plotted with the matplotlib tool (Panu Maalampi).

Chapitre 7

Software maintenance

In this chapter, one will find a short description of DDH routines, and the organization of arrays. In a very practical manner, is described the necessary operations when adding a supplementary field.

7.1 Main arrays and their organization

The main arrays are in two modules :

- YOMTDDH for arrays receiving variables and cumulated tendencies and fluxes.
- YOMMDDH pour the other arrays, except logical variables which are in YOMLDDH.

7.1.1 Arrays describing domains

The distribution of grid-point points in the user's domains is inside NDDHLA (NDLON, NDGL) for zonal bands, NDDHPU (NDLON, NDGL, NDHNPU) for limited domains and isolated points.

- NDHNPU: number of planes used by the user.
- NDDHI (NDLON, NDGL) for the domain distribution,
- HDSF (NDLON, NDGL) for the weights $\sigma_{j,k}$ of each grid-point.

For each (*jlon*, *jgl*) point of the Gauss grid, within NDDHI, one will find the index of the external domain. It varies between 1 and NDHIDH.

- *Global domain*: NDHIDH = 1 every points belong to domain 1.
- *Zonal bands*: NDHIDH = NDHKD
- *Limited areas*: NDHIDH depends very much on declaration details and is very unpredictable.

The following tables allow to reconstitute the user's domains using internal domains.

- NLRDDH (NDHDDX, NDHKD) : integers contained in NLRDDH, from NLRDDH (1, JKD) to NLRDDH (NLXDDH (JKD), JKD), are the internal domains whose reunion makes the latitude band JKD,
- HDSFLA (NDHKD) : weight of each latitude band.
- NURDDH (NDHDDX, 0:NDHBPX, NDHNPU) : integers from NURDDH (1, JDOM, JMASK) to NURDDH (NUXDDH (JDOM, JMASK), are the domains whose reunion makes the user domain (JMASK, JDOM). JMASK is the index of the virtual plane, $1 \leq JMASK \leq NDHNPU$, and JDOM is the index of the domain within the plane: $0 \leq JDOM \leq \underline{NDHBPU (JMASK)}$. Points not assigned by users belong to the domain 0.
- HDSF DU (0:NDHBPX, NDHNPU) : Weight of each user's domain.

The weight of the global domain is HDSFGL, from MODULE/YOMIDDH/.

7.1.2 Data arrays

Pointers contained within MODULE/DDHDIM/and /DDHPON/ from YOMMDDH identify the content of these tables. These tables are HDCV0 (0:NFLEV, NDHCV, NDHIDH, NDHTSK) alias HDCVB0 (NDHCV* (NFLEV+1), NDHIDH, NDHTSK) : arrays at time 0 and tendencies/fluxes cumulated from 0 to NSTEP-1.

HDCV1 (0:NFLLEV, NDHCV, NDHIDH, NDHTSK) alias HDCVB1 (NDHCV* (NFLLEV+1) , NDHIDH, NDHTSK) : variables at time NSTEP and tendencies/fluxes cumulated from 0 to NSTEP.

HDCS0 (NDHCS, NDHIDH, NDHTSK) : soil variables at time 0 and cumulated fluxes from 0 to NSTEP-1.

HDCS1 (NDHCS, NDHIDH, NDHTSK) : soil variables at time NSTEP and cumulated fluxes from 0 to NSTEP.

- NDHIDH : number of internal domains.
- NDHTSK : number of tasks.
- NDHCV : total number of vertical profiles.
- NDHCS : total number of surface fields.

All vertical profiles are defined on NFLLEV+1 words. Generally HDCV(0, field , domain , task)=0. Before des-

cribing these arrays, a few words on *logical pointers*. They are of two kinds

- Permanent pointers who, for each scientific options, show le number of fields of each categories. For the moment, the options are LHDHKS, LHDMCI and LHDENT. Categories are composed of variables, tendencies/dynamical fluxes, tendencies/physical fluxes (i.e. for the moment there is no distinction between tendency and flux).
- situation pointers which depend on chosen options for a given experiment.

Permanent pointers are initialized in SUNDDH. Every counted field corresponds to specific computation FORTRAN code lines INCPDYDDH and CPPHDDH, and to writing lines or other editions in PPEDDH and PPFIDH. A corresponding commentary is in the general nomenclature of YOMTDDH. Situation pointers are also initialized in SUNDDH from permanent pointers

and from logical options given by NAMDDH. They control global lengths in CPG, POSDDH and PPSYDH where fields are undifferentiated.

Permanent pointers are

- NDHVHK : number of variables under LHDHKS
- NDHFHKD : number of fluxes/tendencies under LHDHKS
- NDHFHKP : number of fluxes/tendencies under LHDHKS
- NDHTHK = NDHVHK+NDHFHKD+NDHFHKP

In the same way, for the option LHDMCI

- NDHVMC
- NDHFMC
- NDHFMC
- NDHTMC

For the option LHDENT

- NDHVEN
- NDHFEND
- NDHFENP

Soil (under LHDHKS)

- NDHVS
- NDHFSD
- NDHFSP

(The total number is NDHCS).

Vertical profiles are splitted into categories

$$\begin{aligned} \text{NDHFxxD} &= \text{NDHAxxD} + \text{NDHBxxD} \\ \text{NDHFxxP} &= \text{NDHAxxP} + \text{NDHBxxP} \end{aligned}$$

where (A) stands for tendencies and (B) for dynamical fluxes.

Situation pointers are

- NDHVV : number of variables in vertical profiles
- NDHFVD : number of fluxes/dynamical tendencies = NDHAVD+NDHBVD
- NDHFVP : number of fluxes/physical tendencies = NDHAVP+NDHBVP

The organisation is as follow

Champs	each field has NFLEV+1 levels		
1	instantaneous values	variables	
...			
...			
...			
NDHVV			
NDHVV+1	cumulated values	NDHAVD	dynamical tendencies
...		NDHBVD	dynamical fluxes
...			
...			
...			
NDHVV+NDHFVD		NDHAVP	physical fluxes
NDHVV+NDHFVD+1			
...			
...		NDHBVP	physical tendencies
...			
...			
NDHCV			

For every categories (variables, dynamical tendency, dynamical flux, etc) one may find fields linked to LHDHKS (eventually) then those of LHDMCI (eventually) then those of LHDENT (eventually). Soil arrays work on the same principle. Fields from 1 to NDHVV+NDHFVD are computed in CPDYDDH. Fields from NDHVV+NDHFVD+1 to NDHCV are computed in CPPHDDH.

7.1.3 Main local arrays

Arrays used as liaison between computations in each grid point and partial means in the one side and the overall table, the link between partial means and output on the other side, go into this category.

Arrays for computations at each grid point

ZDHCV (KPROMA, 0 : NFLEV, NDHCVSU) alias PDHCV (KPROMA, NDHCVSU * (NFLEV+1)) fields in vertical profiles in CPG. Alias used in CPCUDDH

ZDHCS (KPROMA, NDHCSSU) soil fields.

— KPROMA : maximum number of points in the horizontal, by grid point task,

— NDHCVSU = max (1, NDHCV)

— NDHCSSU = max (1, NDHCS)

---SU lengths ensure that d'ARPEGE is properly working even if NDHCV=0 or NDHCS=0 (case(s) where DDH diagnostics are not activated).

Arrays for synthesis over a user domain

ZDHCV (0 : NFLEV, NDHCVSU+NDHVV) alias PDHCV ((NDHCVSU+NDHVV) * (NFLEV+1)) fields in vertical profiles. Alias used in PPSYDH.

ZDHCS (NDHCSSU+NDHVS) soil fields.

In this table, only one domain is present at a given time. It is organized as follow

Field	
1	variables at t
...	
NDHVV	
NDHVV+1	tendencies then fluxes
...	dynamical, cumulated
NDHVV+NDHFVD	from 0 to NSTEP-1
NDHVV+NDHFVD+1	fluxes, then tendencies
...	dynamical, cumulated
NDHCV	from 0 to NSTEP-1
NDHCV+1	variables at $t = \text{NSTEP} * \text{TSTEP}$
...	
NDHCV+NDHVV	

ZDHCV array structure.

7.2 Organization of the main functions

An inventory of the main DDH sub programmes and their calling tree is presented here. The following conventions are assumed :

- (—) name of the sub programme between parenthesis : sub programme whose main function is not to compute diagnostics ; generally speaking, sub programme ARPEGE already existing.
- **m** Multitask sub programme.
- **tci** specific DDH sub programme in which all fields are undifferentiated
- **cci** specific DDH sub programme in which every field is specified.

(SU0YOMA) , 0 level initialisations calling

- (SULUN) , initialisation of numbers of file logical units.
- (SUCT0) , initialisation of parameters of output frequencies.
- SUNDDH, initialisation of permanent logical as well as other default values. Read NAMDDH. Deduce dimensions (except NDHIDH and NDHTSK) and logicals.
- (SUALLO) , allocate global arrays.

(SU0YOMB) , initialisations from 0 level, calls

- (SULEG) , computation of Gauss weight $\omega(k)$.
- (SUGEM1) , geographical coordinates computation : (λ_q, μ_q) in each point.
- SUMDDH, verification and set up of domain declarations (from BDEDDH to FNODDH). Distribution of users domains in internal domains, computation of the number of internal domains NDHIDH. Computation of weights of interest for horizontal means (HDSF, HDSFGL, HDSFLA, etc...). Print of computed masks values, by calling PRIMDDH.
- (SUOPH) , generic name of DDH files (CFNDDH from MODULE /YOMOPC/).
- (SUSC2) , computation of the number of logical tasks : $\text{NDHTSK} = \text{NSLBR} - \text{NDGSA} + 1$.
- SUALTDH, allocate global arrays (MODULE /YOMTDDH/). Initialise these to 0.

(CNT1) , level 1 of the model, calls SU1YOM, initialization of output overcontrol : N1DHP and N1DHF (MODULE /YOMCT1/).

(CNT4) , management of the temporal loop, calls (MONIO) , determination of output time steps (IDHFTS, IDHPTS).

(STEPO) , control of the integration at the lowest level, calls

- (SCAN2H) , initialization of the input-output scheme, calls ZERODDH, transfer of fluxes/tendencies cumulated in time from HDCVB1 to HDCVB0 and zeroing of the part of HDCS1 and HDCVB1 tables which will receive the cumulated in the horizontal of variables at the current time.
- (SCAN2M) **m**, multi task interface of grid point computations, calls CPG **m**, grid point computation :
 - Declaration of local arrays IDDHI and ZDHSF (resp. for the domains distribution and for the points weight).

- Declaration of ZDHCV et ZDHCS (fields resp. for 3D and 2D cumulated).
- Interface from NDDHI and HDSF to IDDHI and ZDHSF.
- Call to CPDYDDH [m] [cci], computation in every points of diagnosed atmospheric variables ($\delta p, q\delta p, C_p T\delta p$, etc...), of tendencies and of adiabatic fluxes, and possible call to CPVRDH (if the verification option is activated).
- Call to CPPHDDH [m] [cci], computation in every points of fluxes and of tendencies du to physical parametrizations, soil computation, and possible call to CPVRDH (if the verification option is activated).
- Call to CPCUDDH [m] [tci], partial horizontal mean of variables, stored in HDCVB1 and HDCVB0 if NSTEP=0, if NSTEP different from NSTOP temporal integration and partial horizontal mean of fluxes/tendencies in HDCVB1.
- POSDDH, output management, converts computation of internal domains into users domains, and gives the results on a file or listing.
 - PPVFDH, edition of verifications on a point.
 - PPSYDH [tci], final horizontal means for a user domain, edition of arrays ZDHCV and ZDHCS, multiplication of variables by par $1/(gS_D)$ and of cumulated fields by $(\delta t/S_D)$.
 - PPEDDH [cci], vertical mean budget edition.
 - PPFIDH [cci], writing on file of results of diagnostics for each domain : articles of documentation and fields.
- CPCUVDH, cumulated in time either for a flux or a tendency in case of verification.

7.3 How to add new fields to budgets

This section describes the operations to implement in order to incorporate one or more new fields in the budgets. The sub programmes such as `tc1` will work as long as the dimensions are updated.

Each new field enters into an option (`LHDHKS`, `LHDMCI`, `LHDENT`), and is a variable, a flux or a tendency. In the last two instances, the field may be either of diabatic origin or coming from physical parametrization. Lastly, it can be a soil field. To identify these properties determines the permanent pointers which must be modified, followed by the sub programmes on which to intervene as well as the location of these sub programmes.

1. update of permanent pointer(s). In `SUNDDH`, increment the permanent pointer which corresponds to the option and to the category of the new field(s) (see page 62).
2. Instruction update. Add the field description in `YOMTDDH`.
3. Compute and store the field in `PDHCV` or `PDHCS` in the sub programme `CPDYDDH` or `CPPHDDH`.
4. Add the field(s) to the output file (`PPFIDH`).
5. Add the field(s) to the printed budget, in `PPEDDH`.

7.4 New dataflow for DDH

7.4.1 Introduction

A new coding approach has been proposed in 2009 for extracting diagnostics from the Arome/MesoNH physical parametrisations. It can be used in other parts of the IFS/ARPEGE software. Physical quantities are recorded into a flexible data structure in the parametrisations, and readable by higher level routines. The data structure (a linked list of ad hoc Fortran 90 types) is automatically allocated and indexed as needed by low-level routines, so that physicists can freely choose which quantities they want to record, and how they want to process them. This technical approach greatly simplifies software clarity and maintenance.

Main applications are (1) to provide an easy access to various Arome/MesoNH physical quantities at the level of the physics calling interface and (2) to replace existing DDH in Arpege/Aladin/Alaro if satisfying results are obtained after intensive testing.

7.4.2 Achievements-Future developments

The software is developed progressively and is expected to replace the existing DDH dataflow in the different models after a period of testing. User's feedbacks will be very important to trace potential weaknesses of the present code. Here is the timetable of foreseen code evolutions :

- `cy35t1` : new dataflow available in Arome only for 3D fields. For Arpege/Aladin/Alaro, old DDH structures are kept.
- `cy35t2` : new dataflow can be used in all models (`LFLEXDIA=.TRUE.`) but by default old dataflow is used only in Arpege/Aladin/Alaro. 2D fields are available in the new dataflow.
- 2009 : intensive testing period which expected improvements in the code. Renewing of DDH operators for horizontal averaging may be necessary.
- 2010 : complete switch to new dataflow ? (Would affect IFS code also...)

7.4.3 General basics of the new dataflow

This section describes the content of file `xrd/module/ddh_mix.F90` which contains all the functionalities of the new dataflow. It can be thought as an externalized functionality of the code. New dataflow features are present in the code under the `LFLEXDIA` switch.

Description

The dataflow consists in self allocatable structures similar to GFL but more flexible. This section describes how they are defined, the possible architecture of the code being discussed in section 7.4.5. Each extracted quantity (variable, flux, tendencies...) will be characterized through a Fortran 90 structure type (named here `DDHFLEX`) which defines several attributes corresponding to this quantity.

The structure type named `DDHFLEX` is given here :

```

TYPE DDHFLEX
  CHARACTER(LEN=11)::CNAME !name of field
  CHARACTER(LEN=1)::CFLUX !'F' if flux 'V' if variable 'T' if tendency
  CHARACTER(LEN=3)::CMOD ! 'ARP','ARO': name of model
  LOGICAL:: LKDDH !TRUE if to be stored into DDH
  ! rfield has to be a pointer because allocatable not allowed in structure type
  REAL(KIND=JPRB),DIMENSION(:,:),POINTER:: RFIELD ! value of retrieved field
  INTEGER(KIND=JPIM):: NFIELDIND! position of flux in ddh array
END TYPE DDHFLEX

```

Following attributes are used :

- CNAME is the name of the field as it will appear in the output file. CNAME has to respect the following conventions :
 - First letter has to be either 'F' for a flux, 'V' for a variable or 'T' for a tendency.
 - Second and third letter describes the conservation equation to which the budget applies (see DDH documentation for details) : CT (temperature), QV (water vapour), ...
- CFLUX is a sting that informs about the nature of the quantity stored in the structure :
 - CFLUX='F' for a flux
 - CFLUX='T' for a tendency
 - CFLUX='V' for a variable
- CMOD gives information on the model's name
 - CMOD='ARO' for AROME
 - CMOD='ARP' for ARPEGE, ALADIN and ALARO (by default but if you wish other labels can be introduced)
- LKDDH is a flag set to .TRUE. if the field has to be processed by DDH operators and to .FALSE. otherwise.
- RFIELD is a pointer corresponding to the value of the field (it will be explained later why it has to be a pointer)
- NFIELDIND is an integer that gives the number of the processed field within the list of all fields.

These attributes are important because they document the structure content itself (important for debugging purposes) and they determine which operations the extracted field will undergo at the place where it is recorded, before being stored (for instance conversion from potential temperature to temperature...)

The various extracted fields are gathered into an allocatable array of structure of type DDH, called here RDDH_DESCR and whose last dimension corresponds to the total number of extracted fields :

```

TYPE (DDHFLEX),ALLOCATABLE,DIMENSION(:):: RDDH_DESCR

```

The attribute *allocatable* being forbidden inside a type structure, the field is not directly stored inside RDDH_DESCR but defined through a pointer to a large array called RDDH_FIELD :

```

REAL,DIMENSION(:,:,:),ALLOCATABLE,TARGET::RDDH_FIELD ! target of RFIELD
! first two dims are the same as PFIELD, the third being the number of stored fields

```

Extracting a field from the physics

For adding a field into the diagnostics, you only need to call subroutine ADD_FIELD_3D and that's all ! The first argument of ADD_FIELD_3D will be the field to store and the others will correspond to the associated attributes (for instance "call ADD_FIELD3D(field_to_store,'name_of_field','F','CT'....)")

Arguments of ADD_FIELD_3D(PMAT,CDNAME,CDFLUX,CDMOD,LDINST,LDDH) are the following :

- PMAT : the array to be stored. It has to be with levels in the same order than in Arpege part of the code. If you are in a .mnh subroutine just use subroutine INVERT_VLEV.MNH before calling ADD_FIELD_3D in order to have levels ordered as in Arpege.
- CDNAME : name of field. It is constructed the following way :
 - CDNAME(1) : 'F' if flux , 'T' if tendency, 'V' if variable
 - CDNAME(1:2) : type of variable ('CT','QI','QV','QR',...)
 - CDNAME(3 :) : name of process
- CDFLUX : 'F' if flux , 'T' if tendency, 'V' if variable
- CDMOD : 'ARO' if AROME, 'ARP' otherwise (but you may add some other label if you wish)
- LDINST : 'TRUE' if instaneous field
- LDDH : 'TRUE' if field is stored to be in DDH

When using `add_field_3D` it is extremely important to have the right attributes in the right order. So be careful! Have a look at `xrd/module/ddh_mix.F90` if any doubt.

Here are some examples :

```
CALL ADD_FIELD_3D (ZTMPAF, 'VQI', 'V', 'ARP', .TRUE., .TRUE.)

CALL ADD_FIELD_3D (ZTMPAF (:, :), CLNAME, 'T', 'ARP', .TRUE., .TRUE.)

CALL ADD_FIELD_3D (PFRSO (:, :, 1), 'FCTRAYS0', 'F', 'ARP', .TRUE., .TRUE.)
```

`ADD_FIELD` performs the following tasks :

- when in the code a specific field is supplied as argument for the first time in the execution, the last dimension of arrays `DDH_FIELD` and `DDH_DESCR` is incremented in order to add space for the new field to store. The code determines if a field is encountered for the first time by testing the field's name. This reallocation of arrays may slow the code and fragment memory during the first time step, but it avoids going through complicated setups. One could also preallocate the arrays according to a first guess of the dimensions, as chosen by the user.
- at every time step the field is stored in `RDDH_FIELD` through the pointer `RFIELD`
- at every time step, some transformations are done on the field according to its nature (and documented by its attributes), for instance conversion from θ to T ... These operations also depend on the physics used (Meso-NH, Arpege...). Here it will be possible for users to add parts corresponding to specific needs, and to document them through attributes.

7.4.4 Activating and modifying the new dataflow

Using DDH products included in documentation

The DDH documentation holds as a reference for the formulation of the budget equations and for the list of terms present by default in the DDH files. If you just need these products, just set the DDH namelist according to your need and you just have to plot the ddh files using the `ddhtoolbox`. In Arome, new dataflow is activated by default. For Arpege, Aladin and Alaro you have to set `LFLEXDIA=.TRUE.` in namelist in order to use the new dataflow. Otherwise old dataflow is used. We recommend to use the new dataflow since old dataflow is kept for the moment only for compatibility with ECMWF and validation purposes.

Adding terms to the already existing DDH products

You just have to call `ADD_FIELD_3D` (Make sure that you have imported this function by adding in your file `USE DDH_MIX, ONLY :ADD_FIELD_3D`) If you want to add a term to an existing budget equation, just use the same name for the variable ('CT', 'QR'...) than in the rest of the code. Otherwise you are free to introduce a new name. If you are in a `.mnh` subroutine, you have to proceed in two steps :

- First you have to transform your array on `NLEV+2` levels to an array on `NLEV` levels in reverse order (to go from the "MNH" word to "Arpege" word). There is a subroutine dedicated to this transformation `INVERT_VLEV.MNH`
- Then use `ADD_FIELD_3D` on the transformed array.

Using the dataflow for extracting terms from the physics but not for DDH

It is possible by just setting `LDDH` to `.FALSE.` when calling `ADD_FIELD_3D` to use the flexible dataflow for retrieving fields out from the physics and use them elsewhere. Once the field is stored using `ADD_FIELD_3D`, you just have to go through the flexible structure once to have the index `MYINDEX` of your field that you can use later on by accessing `RDDH_FIELD(:, :, MYINDEX)` :

```
DO II=1, NTOTFIELD
  IF (RDDH_DESCR (II) \%CNAME=='MYNAME') THEN
    MYINDEX=RDDH_DESCR (II) \%NFIELDINF
  ENDIF
ENDDO

% your field is stored in RDDH\_FIELD(:, :, MYINDEX)
```

For the time being the previous lines of code are not in the common cycles, if you feel that there should be just send an email to the DDH team.

Miscellaneous

If the budget package in Méso-NH is maintained (BUDGET routine) there is nothing to do in the DDH part of the code, except in the following situations :

- **New species are added in Arome.**
 - In this case, a label for it first has to be introduced.
If this is an hydrometeor you have to add an entry to CLVARNAME in APL_AROME (it corresponds to the names of hydrometeors ordered the same way than in PTENDR) and report it coherently in MODDB_INTBUDGET. Increase also by one dimension of TAB_VARMULT array. **Beware to use the same ordering of variable than in Méso-NH calls to budget!!!** If this is not an hydrometeor, it may not be present in the Méso-NH budgets and thus we recommend to use combination of INVERT_VLEV and ADD_FIELD_3D.
 - The transformation applied to this field has to be defined.
In ARO_SUINTBUDGET, increase by one the last dimension of TAB_VARMULT and have it pointing on the TCON2 (equal to PQDM) since it is an hydrometeor.
 - In APL_AROME, check that loops on last dimension on PTENDR include this new hydrometer.
 - In CPDYDDH just use ADD_FIELD_3D to add the value of the variable corresponding to your new hydrometeor.
- **Order of subroutines is changed in APL_AROME.** In this case make sure that ARO_STARTBU and ARO_SUNINTBUDGET are called at the right place.

7.4.5 Architecture of the code

Subroutine ADD_FIELD_3D and associated modules are in xrd/module/ddh_mix.F90. This subroutine contains all elements for using the new dataflow.

However the use of the new dataflow in the part of the Arome code originating from Méso-NH required some interfacing described in the following subsection.

Calling tree

Example : correction of negative QL values by the AROME physics.

- The correction is done in aro_rain_ice.mnh (in mpa/micro/externals).
- To activate the DDH budget of the QL tendency, due to this correction, aro_rain_ice.mnh calls BUDGET :

```
IF (LBUDGET_RC) CALL BUDGET (PRS(:, :, :, 2) * PRHODJ(:, :, :), 7, 'NEGA_BU_RRC')
```
- The routine BUDGET (in mpa/micro/internals) used for AROME runs differs from that used for MNH runs. The AROME BUDGET routine converts the MNH name 'NEGA_BU_RRC' into the DDH name 'TQLNEGA', and then calls ADD_FIELD :

```
IF (CLPROC/= 'INIF') CALL ADD_FIELD_3D(ZVARS, CLDDH, 'T', 'AROME', LINST, LDDH)
```
- ADD_FIELD_3D (in xrd/module/ddh_mix.F90) allocates the relevant arrays if not already allocated, and then writes the real data of the tendency into the RDDH_FIELD array from module ddh_mix.

Organization of the data flow

The DDH diagnostic facility performs some domain averaging and budget computation after the diagnostic extraction. These operations are performed at each timestep, after the physics computations, so that the raw recorded fields are accessible as NPROMA packets at the level of APLPAR/APL_AROME, where they may be used for other purposes.

For the DDH domain averaging, the Arpege subroutine cpcuddh.F90 (see DDH documentation for more details) is used and averaged fields are then written into file in pppidh.F90 (which will be simplified since now with the self-documented structure, a loop on all elements in DDH_DESCR can generate the names of the fields to be written into the DDH file). The subroutine cpcuddh.F90 uses arrays (hdcvbx stored in module yomtdh) whose size is computed in setups (the total number of fluxes/tendencies depend on the options used for physics). Since these setups are no longer used with the new data flow, these arrays are allocated with an estimated size (larger than expected value) for the time being but we are thinking at a way to have them reallocated or initialized elsewhere in the code after a dummy call to the code that only computes the total size of DDH arrays (like the call to stepo from cnt4.F90 if CFU/XFU diagnostics are switched on).

Figure 7.1 summarizes the new data flow (which is the same for Arpege and Arome) within a time step.

Application to DDH in Arome

The new dataflow is used in Arome since cy35t1 for DDH diagnostics. Méso-NH code already uses its own diagnostics through the sophisticated budgets and advantage is taken from the work already performed there in order to avoid duplication

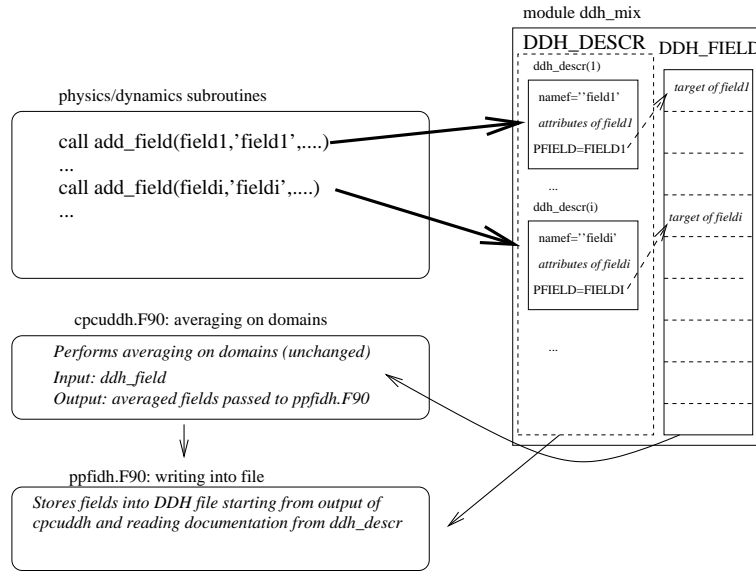


FIGURE 7.1 – Organization of the data flow within a time step. Subroutine `ADD_FIELD` stores the field and the associated description into `DDH_DESCR` after possible transformations (bold arrows). Averaging on the domains is performed as in Arpege in `cpcuddh.F90`, the output being written into file in `ppfidh.F90` using the description of the fields stored as attribute in `DDH_DESCR`.

of effort. MNH's budgets are called through the call of the subroutine `budget` after each process. This subroutine is able to perform operations on the stored quantity. In order to keep the maximum level of compatibility between MNH and Arome code, it was chosen to keep the calls to `budget` unchanged in the Méso-NH code and to write a new budget subroutine that would be called in Arome instead of the `budget` from `MASDEV`. This subroutine, located in `/mpa/micro/externals`, suppresses first and last level of MNH fields and reverses the order of the vertical levels.

In Arome there are two different ways to have terms in the DDH products. The first is to use `ADD_FIELD_3D` after a call `INVERT_VLEV` as shown previously. The second possibility is to use the budgets from Méso-NH as in the first version of the DDH code in Arome. We have used a combination of the two methods in order to take advantage of the validation performed by the Méso-NH team of the budget packages.

- Variables are stored in `cpdyddh.F90` using `ADD_FIELD_3D` since the part of the code is common with Arpege/Aladin/Alaro.
- Within `APL_AROME`, adjustment and radiation are retrieved using `ADD_FIELD_3D` and other processes through budgets from Méso-NH.

Interfacing with Méso-NH budgets works the following way :

- `ARO_SUINBUDGET` stores quantities (Exner function...) necessary to transform tendencies from Méso-NH (in θ, r variables) to tendencies in (T, q) variables into the module `MODDB_INTBUDGET`
- `ARO_STARTBU` stores initial values of tendencies for each variable
- Within Méso-NH, subroutine "BUDGET" is called. The `BUDGET` subroutine from Méso-NH is replaced by a new subroutine (`/mpa/micro/internals/budget.mnh`) called the same way with the same arguments that transforms tendencies of Méso-NH variables ($\theta...$) to tendencies on the desired variables ($c_p T...$) and skips the Méso-NH processing.

7.4.6 Remaining issues specific to the new dataflow

Some issues are still to be dealt with in the new dataflow :

- Performances. If faster in Arome than the old code, there is still room for improvement in terms of computational performances.
- OpenMP. The code has to be tested and optimized for OpenMP parallelization. For the time being, validation has only been done on the NEC platform from Météo-France.
- Elarging the flexibility of the code to the DDH operators for domain averaging. Some arrays like `PDDHCV_TOT` still have to be initialized at the beginning of the time step and thus we don't fully benefit from the flexibility of the new dataflow. Thinking about how to deal with that without affecting the part of the code used by ECMWF is ongoing.

7.4.7 Conclusion

This new version of the dataflow offers not only more facilities to add new quantities in the diagnostics but also more flexibility in terms of possible uses of these diagnostics. For developers, since the new code is considerably smaller and readable than the current one in Arome, it will be easier to debug and maintain when physics evolve in the future. We also expect an increase in the code's performance for Arome's DDH since the Meso-NH budgets part of the code (with a lot of unused (in Arome) options slowing the code) will be skipped.

Another important aspect is that this tool, after being successfully implemented in Arome can now be used in Arpege/Aladin/Aladin. Before going on with further work to upgrade this prototype version towards a beta version, discussion between the different possible users of this type of diagnostics is needed in order to raise possible new issues and needs regarding what different users would like these structures to offer.

Chapitre 8

History

- 1991 :** Initial analysis and coding of the DDH software (Alain Joly).
- 1992 :** Introduction of entropy and kinetic energy budgets (Martin Janousek, Jean-Marcel Piriou).
- 1993 :** Introduction of relative humidity, liquid and ice water (Jean-Marcel Piriou).
- 1997 :** New surface fields, IFS model (Pedro Viterbo).
- 2000 :** Diagnose surface "tiles", IFS model (Christian Jakob).
- 2006 :** Extract AROME physical data flow, interface to DDH routines (Tomislav Kovacic).
- 2006 :** Write AROME DDH documentation (Tomislav Kovacic).
- 2007-11 :** Rewrite the budget tool "ddhb", still based on the "ddhi" and "ddht" existing ones (Alex Deckmyn, Jean-Marcel Piriou, Tomas Kral).
- 2008-07 :** Draft translation of the French DDH documentation into English, by Jean Maziejewski International Sekretarski, approved by Jean-Marcel Piriou.
- 2008-07 :** Create the ddhtoolbox, write its documentation (Jean-Marcel Piriou).
- 2009 :** Interface AROME physics with DDH, new flexible dataflow for AROME, ALADIN and ARPEGE (Olivier Rivière).
- 2018 :** Dynamical DDH : semi-lagrangian advection, horizontal diffusion, semi-implicit (Fabrice Voitus).
- 2019 :** Flexible DDH : adding a new field in calling NEW_ADD_FIELD_3D (Fabrice Voitus).
- 2023 :** A new section was written in the ddhtoolbox documentation, to learn step by step, from examples, the use of the DDH tools (ddhi, ddhr, ddht, etc) (Jean-Marcel Piriou).
- 2023 :** The dd2gr plot tool is now part of the ddhtoolbox, to allow an easier start with DDH use (Jean-Marcel Piriou).