# The AROME prototype : Technical documentation CY29T2

Yann SEITY (CNRM/GAMP), Sylvie MALARDEL (CNRM/GMME)
and Gwenaëlle HELLO (CNRM/GMAP)
Third version, April 2005.

## 1.Basic considerations

The AROME prototype is built upon the dynamical kernel of ALADIN-NH. The physical package is imported from the research model Méso-NH (GMME-LA). The two worlds are separated (Figure 1). The communication between the two worlds are performed thanks to interfaces that were written for the prototype.

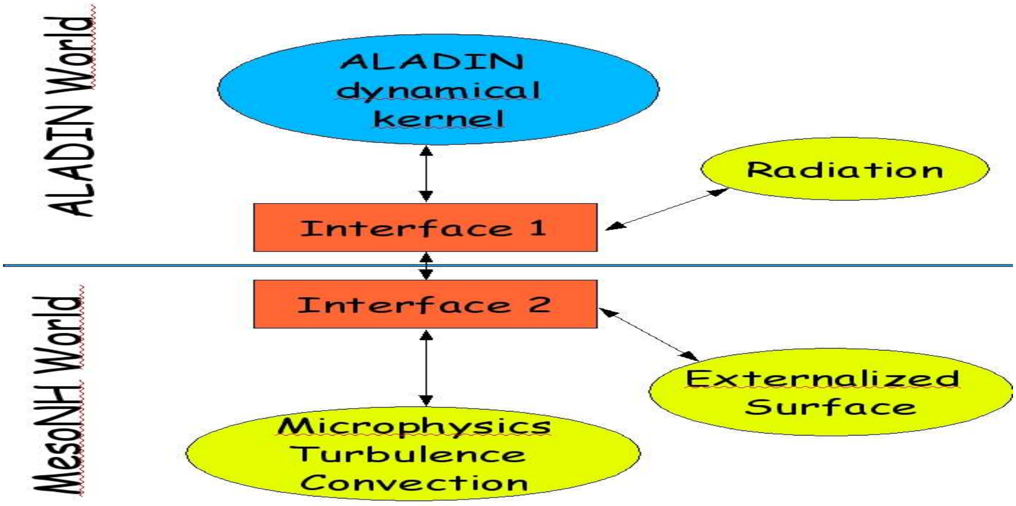

Figure 1 : AROME prototype organisation

### Méso-NH world

The routines of the physical parametrisations are the ones of Méso-NH model (MASDEV47bf), without any major change. Some minor changes were necessary in 5 MesoNH routines (turb,turb_ver_dyn_flux, tke_eps_sources, average_rad and modd_parameters). These modifications will be included in the next MesoNH cycle under a specific AROME key declared on MesoNH.

**ALADIN world**

The prototype, developped on pre_cycle 27has been phased on the main version of Cycle 29T2. The call of the mesoNH physics is done in Interface 1 (Figure 1) with apl_arome (which replace aplpar in the AROME world). For the prototype, we have created some new routines corresponding to the interface1 part of Figure1 (cf the list in part 6.1). We also modified some ARPEGE/ALADIN routines (most of them to update the R and Cp calculations according to new microphysics ).

## 2.Setup

The specific setup of AROME is done in 3 places in the code (Figure 2) :

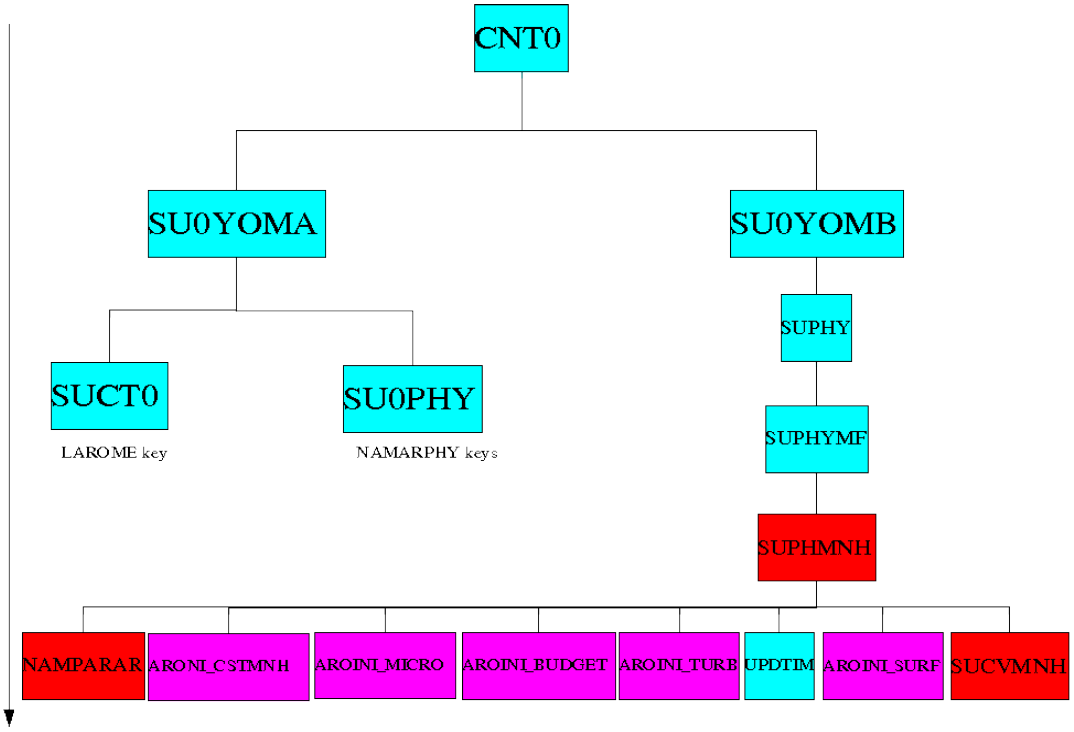

Figure 2 : AROME setup organisation. Arpège-Aladin routines are represented in blue, Interface1 routines in red, and Interface2 routines in pink

1)When reading NAMCT0 in SUCT0, the LAROME key is initialized. This key is T in case we run AROME

2) In su0phy, the NAMARPHY namelist is read. It contains the key LARPHY which controls the calling of the AROME physical interface, and 4 other keys for each individual physical parameterization (LMICRO, LTURB, LGROUND and LKFBCONV for AROME 10km). As AROME used the ECMWF radiation code, which can also be called by ALADIN , the key for calling the radiation scheme (LRAYFM) already exists in NAMPHY and so, it is not dupplicated in

NAMARPHY.

3) Until now, we only initialized namelist keys. According to these keys, we now have to call MesoNH physical setup routines. This is performed through interafce2 routines : aroini_cst_mnh for physical constants, aroini_micro for the microphysics, aroini_turb for the turbulence and aroini_surf for the surface scheme. These routines are called by suphmnh, under suphmf. suphmnh also calls sucvmnh in AROME 10km case for the initialisation of some convection scheme variables, stored in yomcvmnh. Suphmnh and sucvmnh are in the interface1 part of figure1.
Suphmnh also read NAMPARAR namelist which is a specific namelist devoted to optional configurations of AROME (academic 2D squall line case, prints for debugging, diagnostic of Wmax).

AROME setup variables are declared and storred in a specific module named YOMMNH.

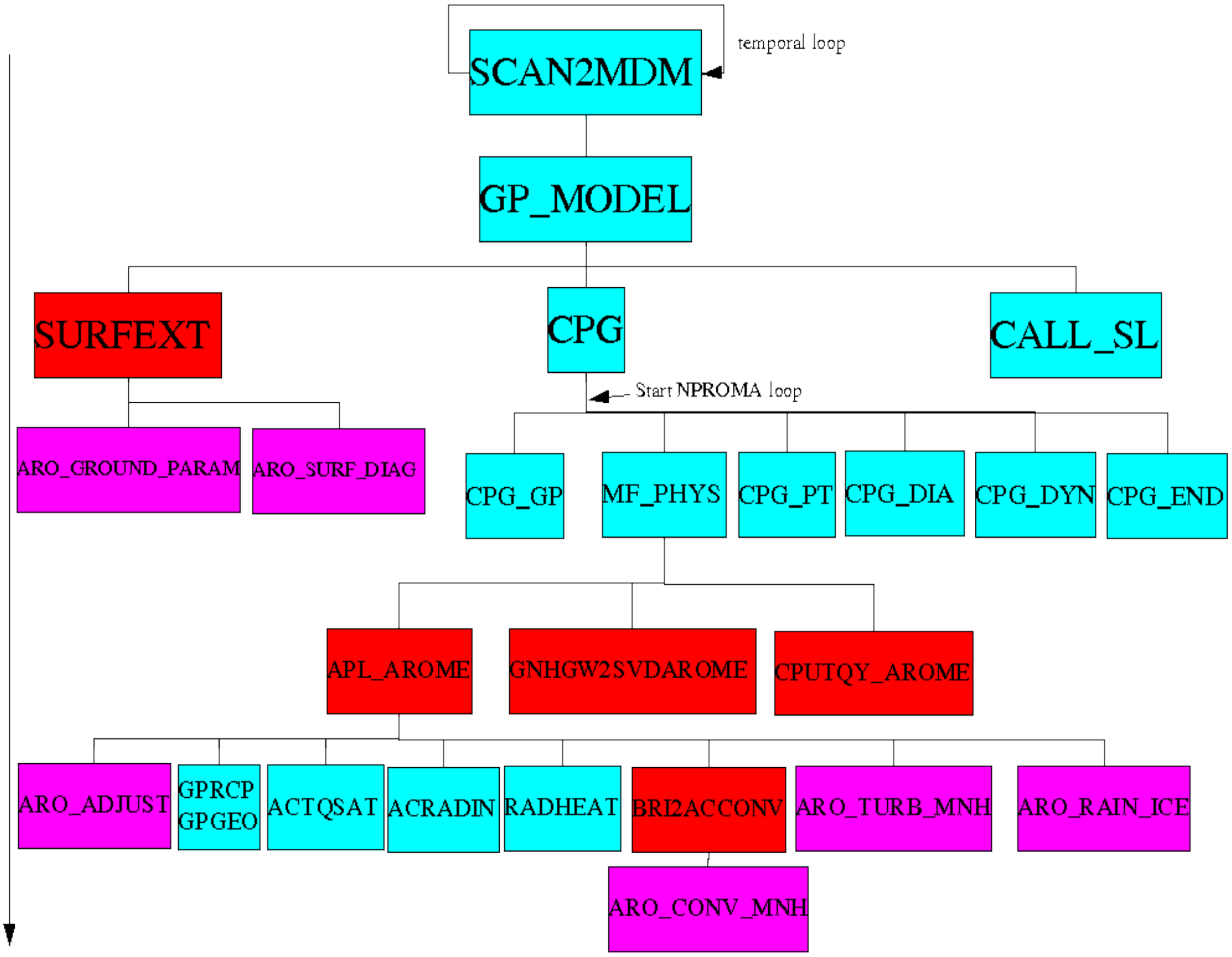

Figure 3 : Integration of AROME Physics in the ALADIN time step. Arpège-Aladin routines are represented in blue, Interface1 routines in red, and Interface2 routines in pink.

AROME needs new prognostics variables, compared with ALADIN, which are GFL type variables :
CLOUD_WATER, ICE_CRYSTALS (already existed and used at ECMWF)

RAIN , SNOW, GRAUPEL, TKE (Turbulent Kinetic Energy) (Arome specific variables)
It also needs 3D pseudo-historic variables wich are of GFL type named :
CLOUD_FRACTION (already exists and used at ECMWF), and SRC (sigma for subgrid condensation).

## 3.Time step organisation and data flow modification

The calling of the AROME physics is done in 2 places in the code : via apl_arome, and via surfext (Figure 3).

apl_arome is the main AROME routine which calls all the physical parameterisations (except surface) :

radiation via ACRADIN, 1D turbulence via ARO_TURB_MNH, microphysics via ARO_RAIN_ICE and convection via BRI2ACCONV. Contrary to the ALADIN physics which computes fluxes, the mesoNH physical parameterisations give directly tendancies of prognostic variables.

APL_AROME is called under mf_phys (like aplpar for ARPEGE/ALADIN physics), under the LARPHY key.

GNHGW2SVDAROME, convert w tendancy in vertical divergence tendancy.

After apl_arome, under mf_phys, CPUTQY_AROME performs the job done by CPUTQY in ARPEGE/ALADIN, adding the physical tendancies to SL buffers. In a future step, these two routines would be merged.

The previous version of the externalized surface was not able to run with NPROMA packets, like it is done in the ARPEGE/ALADIN physics. For AROME, the surface is called outside of the NPROMA loop, under GP_MODEL (like it is done in IFS for radiation), with a new interface named SURFEXT. At the first time step, the surface is not called (because radiation fluxes are still not calculated) and fluxes are null in input for the turbulence. For the next time steps, fields necessary in input for surface scheme are stored in apl_arome in a buffer (named GPARBUF), and recovered under surfext. Fluxes given to the turbulence are those calculated in the forward time step by the surface. In the last version of the surface code (MASDEV47bf), the deletion of the $n and its replacement with pointers structure allow us, with minor coding, to put back the surface call inside the NPROMA loop. The code has not been developped yet. Under surfext, ARO_GROUND_PARAM call the surface code, and ARO_SURF_DIAG control the write of surface historic files.

## 4.Where does AROME impact in the arp/ald code ?

A lot of routines were modified only because new microphysics change the calculation of R and Cp and each routine calling GPRCP had to be modified to add 3 new arguments (Rain, Snow and Graupel). It will be included on CY29T2 and should not change with AROME (except if we add hail in a future version). These routines are not defined below.

There are 30 routines modified for AROME not only for GPRCP calls :

**arp/adiab/cpg.F90** : new arguments to mf_phys, cpg_gp and cpg_end, cpg_pt+w calculations+do not compute diagnostics in case LARPHY.
**arp/adiab/cpg_gp.F90 :** Read PGPAR in GPARBUF (under LARPHY key)
**arp/adiab/cpg_end.F90 :** Write PGPAR in GPARBUF (under LARPHY key)
**arp/adiab/cpg_pt.F90 :** Add new GFL pronostic variables
**arp/adiab/cpphinp.F90 :** Do not compute MOCON in LARPHY case (impossible because gridpoint Q)
**arp/adiab/lavent.F90 :** under LSQUALL key, for 2d academic squall line, modify wind for advection.

**arp/control/gp_model.F90 :** Call Surfext under LARPHY and LGROUND keys.

**arp/module/yom_ygfl.F90 :** Define AROME GFL
**arp/module/yom_ct0.F90 :** add LAROME key
**arp/module/yom_fa.F90 :** define YFA names for AROME fields (under LAROME key)
**arp/module/yomgppb.F90 :** define GPARBUF
**arp/namelist/namct0 :** add LAROME
**arp/namelist/namfa.h :** add new AROME fields
**arp/namelist/namgfl.h :** add new AROME GFL
**arp/phys_dmn/acradin.F90 :** 4 new arguments for AROME (needed by externalized surface)
**arp/phys_dmn/mf_phys.F90 :** New arguments for AROME+call apl_arome under LARPHY key.
**arp/phys_dmn/recmwf.F90 :** idem acradin
**arp/phys_dmn/suphmf.F90 :** Call suphmnh under LARPHY key. (stay under the key ???)
**arp/setup/su0phy.F90 :** default values for namarphy+switch off arp/ald physical keys in case LAROME.AND..NOT.LFPOS+read namarphy.
**arp/setup/suct0.F90 :** LAROME key defautl value (F)
**arp/setup/sudim1.F90 :** default values for new GFL_NL (exemple YTKE_NL). LAROME key is only used for a test (all AROME GFL in gridpoint) which will call abor1 if not satisfied.
**arp/setup/sudyn.F90 :**Call set_gfl_attr for new GFL (SRC)+sous clé LAROME :
a) initialisation de LADVAMV à T si LADV, F sinon.
b) CALL SET_GFL_ATTR(YA,LDADV=YA_NL%LADV au lieu de LADVAMV pour ne pas advecter YA (F en namelist))
c) CLSLINTAROME='LAITQM au lieu de CLSINT pour la TKE afin d'éviter la création de valeurs négatives.
**arp/setup/sugfl.F90 :** define_gfl_comp for new AROME GFL
**arp/setup/sumpini.F90 :** use LAROME for namct0 read.
**arp/setup/sufa.F90 :** define name of GFL used in AROME (under LAROME key)
**arp/setup/sugridua.F90 :** special case of TKE in AROME. All is under LAROME key.
**arp/phys_ec/radlsw.F90 :** 4 new arguments for AROME (cf acradin and recmwf).

**arp/phys_ec/sw.F90 :** idem
**arp/phys_ec/sw1s.F90 :** idem
**arp/phys_ec/swni.F90 :** idem

## Namelists

In an ARPEGE/ALADIN/AROME namelist file, two namelists are specific to AROME : namarphy and namparar.

In namarphy :

LARPHY=true, to run arome physics
LMICRO=true, to use ICE3 microphysics scheme
LTURB=true, to use 1D turbutence scheme
LGROUND=true, to use surface scheme
LKFBCONV=true, to use Kein-Fritsch-Bechtold convective scheme (used only in AROME 10km )
You can run microphysics alone. You can also run microphysics with turbulence in case you also provide idealized surface fluxes. But if you want to use radiation scheme (LRAYFM=T in namphy), you have to switch on surface scheme to have good values of albedo and surface emissivity. If radiation is switched on but surface scheme not, it will run with the constant albedo and emissivity which were on the initial file as in ALADIN. The surface scheme cannot run alone. You have to switch on, radiation and microphysics to compute good fluxes.

In namparar :

LOSUBG_COND and LOSUBG_AUCV are options for subgrid condensation scheme. They are fixed to F by default.
NSWB_MNH is the number of spectral bands in input radiation needed by surface scheme. It is fixed to 6 by default in suphmnh.
NPRINTFR is the frequency of prints in apl_arome. By default, it is fixed to 36 hours, but if you need prints for debugging for instance, putting NPRINTFR=2 will produce prints one time step over two.
NPTP is the number of the point in the NPROMA pack on witch the prints are performed.
LSQUALL,NREFROI1,NREFROI2 and VSQUALL are only used for the academic 2D squall line case simulation. Lsquall=F by default. Nrefroi1 and Nrefroi2 allows to fix the area of the initial cooling.Vsquall is the speed of the moving squall line, to counterpart in the advection in order to plot results in a referential moving with the squall line.
NDIAGFR is the frequency of surface ouput files. If negative, no surface diagnostics are performed. Otherwise, a file is produced by the surface every NDIAGFR time step by a call to aro_surf_diag under surfext.
LDIAGWMAX activates a print of the maximal vertical velocity on each nproma pack every NDIAGWMAX time step.

In the other namelists, we have

LSPRT=T in namct0 for AROME in order to run with gridpoint Humidity.
LAROME=T in namct0.

The GFL attributes are in NAMGFL , as an example for cloud water :

```
 YL_NL%LADV=.TRUE.,
 YL_NL%LREQIN=.TRUE.,
 YL_NL%LGPINGP=.TRUE.,
 YL_NL%LGP=.TRUE.,
 YL_NL%LSP=.FALSE.,
 YL_NL%LCDERS=.FALSE.,
 YL_NL%LT9=.FALSE.,
 YL_NL%LT1=.TRUE.,
 YL_NL%LT5=.FALSE.,
 YL_NL%LSLP=.FALSE.,
 YL_NL%LCOUPLING=.TRUE.,
 YL_NL%LPT=.TRUE.,
```

LRRTM=T in NAERAD to use RRTM long-wave radiation code.
To use climatologic aerosols, you have to put NAER=1 in NAERAD and NVCLIA=4 in NAMDPHY. The way of initialisation is the same as in Aladin. Tegen or Tanre fields have to be added in the clim file produced by the E923 configuration.
To use Ozone climatologies, two ways are actually coded : As it is done in Aladin (NOZOCL=2 in NAERAD and LO3ABC=T in NAMPHY) or as it is done in ECMWF (NOZOCL=1 in NAERAD ).

LMPHYS=T in NAMPHY to enter in mf_phys, but all the other keys of NAMPHY are set to F, except LRAYFM (tu run Fouquart-Morcrette SW radiation code) and LO3ABC if Ozone uded as in Arpege/Aladin.

## 5. Input/Output files.

Here is presented the present status of the I/O files for the prototype. It might change in the future versions of the prototype.

Input :

The AROME prototype needs two kinds of files in input :
a) MesoNH file (Initial Condidion (IC) only) containing surface data needed by the externalized surface scheme. For the moment, it is still prepared with MesoNH tools, but in the future (next 3 months), it will be possible to create it during E927 with specific AROME tools.
b) Aladin FA files (IC and coupling) containing atmospheric fields, surface geopotential and surface pressure. In the first version of the prototype, these files were produced with Mesonh tools and converted in ALADIN files with 'convmnh2ald' arotool. Now, we are also able to prepare them with E923 and EE927 or E927 ALADIN configurations.

Output :

AROME also produces two kinds of output files :
Mesonh files for surface diagnostics (if required by NDIAGFR key in namparar). This files also contain cumulated and instantaneous surface rainfalls (rain, graupel and snow). As any mesoNH file, it is plotted with diaprog.
Aladin historic files, containing atmospheric fields. It can be plotted either with chagal or metview. It is also possible with diaprog, after being converted in a Mesonh file with the 'ald2mnh' arotool.

# 6.Source architecture on Clear Case for the new routines :

Clear Case software is available on andante. Documentation is available on the GMAP web site.

## 6.1Routines under arp vob

It concerns the routines written in Interface 1 (Figure 1) for AROME.
You can find their location and name on the following Figure.

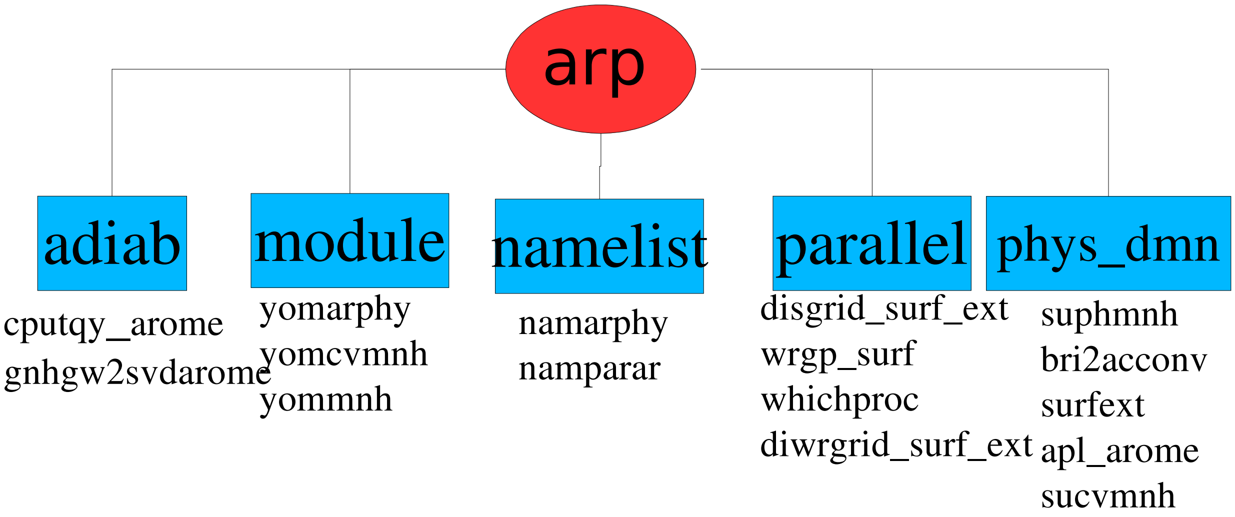

Figure 4 :Name and location of AROME Interface1 routines under arp Vob on Cear-Case

## 6.2MesoNH routines and Interface 2 routines

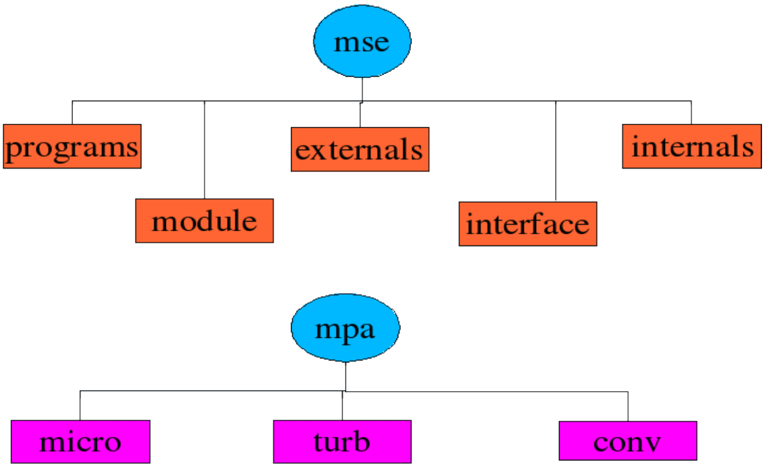

Figure 5 : Clear-Case tree of MesoNH routines

In the MesoNH community, the sources are stored and managed with RCS software. For Arpège/Aladin, in GMAP and COMPAS/GCO, the sources are managed with Clear-Case. For practical reasons, the mesoNH sources used in AROME were introduced under Clear Case. Two new Vobs have been created to store them and the interface parts developped for AROME. The Vob named 'mse' (MesoNH Surface Externalisée) contain the externalized surface part (which will be used also in Arpège/Aladin in the future). The Vob 'mpa' (MesoNH Paramétrisations Atmosphériques) contains microphysics, turbulence, and convection parts. The tree is the same under micro, turb, and conv than it is under mse. It is copied from what have been done at ECMWF for externalized transform packages (Vobs 'tal' and 'tfl'). You will find the 'pure mesoNH' code under 'internals' directory whereas the interface2 routines developped for AROME are under the 'externals' one. Because of compilations constraints, all the modules are stored under 'module' directory. 'Interface' contains interfaces of the routines under 'externals'. 'Programs' is not fullfilled for the moment. It could contain in the future test programs of each individual package.

The specification of the kind of real and integer variables is not done in the same way in MesoNH and in Aladin. In Aladin, a kind is explicitly defined (in parkind1.F90). It is not the case in MesoNH. For the compilation, the same options will be used between MesoNH and Aladin parts except the fact that for routines under mse or mpa, the real double precision compilation option has to be added (to be consistant with the compilation of arp/ald sources). For this reason, the name of sources under mse or mpa are not named .F90 or .f90, but .mnh.

Further developments on the AROME prototype have to be done on Clear-Case.

# 7. Compilation/development

The compilation of the AROME code is done as for ARPEGE/ALADIN with the 'gmkpack' software (documentation on the GMAP web site) on tora.
You have to use the latest version of gmkpack (named 6.1) witch will become the default version by the end of April 2005.
To try right now this new version you should export in your .profile on tora :

```
export GMKROOT=/u/gp/mrpm/mrpm602/gmkpack.6.1
export GMKTMP=$TMPDIR
```

To create a binary on cycle 29T2, you have to type on tora :

```
 gmkpack -r cy29t2  -u name_of_your_pack  -p arome
```

And to submit the ics_arome script.
If you only want to use the official AROME binary of CY29T2 without any modification in the code, you do not need to use gmkpack. GCO have compiled it for you. You can retrieve it in a script by :

```
gget  al29t2_arome-main.01.L0209.x.exe
```