# INTERFACES DE MANIPULATION DES FICHIERS ARPEGE/ALADIN

**J. Clochard, R. El Khatib, D. Paradis**

**Version de documentation remaniée en octobre 95 puis en Novembre 2000, correspondant au cycle 22 (hiver 1999/2000), avec référence aux extensions pour le cas "domaine limité" (ALADIN, Full-POS).**

**Dernière mise à jour le 04/09/02 par R. El Khatib pour le cycle 24**

## TABLE DES MATIERES

# 1) GENERALITES SUR LES DONNEES ENREGISTREES

## 1.1) Fichiers ARPEGE/ALADIN

Par **"fichier ARPEGE/ALADIN"** on désigne le **principal type de fichier servant d'intermédiaire entre programmes de la chaîne ARPEGE/ALADIN**, à savoir:

- ébauche et résultat (brut) d'analyse;

- situation de départ (démarrage seulement) et sortie (brute) du modèle de prévision, ou de son initialisation;

- entrée et sorties de post-traitement destiné à alimenter une base de données (BDAP) ;

- entrée et sortie de post-traitement destiné à produire les fichiers de conditions aux limites permettant d'executer le modèle ALADIN ;

- résultat de préinitialisation, ou plus généralement d'un programme visant à utiliser des données externes au projet ou d'une configuration modèle différente;

- champs de constantes et de climatologie;

- éventuellement, moyennes ou cumuls temporels.

## 1.2) Structure des données

La **structure** de ces fichiers est **indexée par nom,** utilisant le logiciel de fichiers indexés LFI.

Il y a **un** article logique par champ horizontal, codé implicitement dans un format de type GRIB. **Généralement, les unités S.I. sont utilisées bien que ce ne soit pas nécessaire.** Il y a possibilité de ne pas coder les champs , voire de les coder différemment (ultérieurement).

GRIB, code OMM (édition 0), est utilisé ici plutôt comme un moyen de coder l'information "champ" sous une forme compactée, une partie des aspects autodocumentation de GRIB étant utilisée pour des contrôles. La partie GRIB n'est pas accessible en tant que telle, seuls des descripteurs le sont.

Un **champ horizontal** est représenté soit en valeurs points de grille **consécutives** sur la grille de collocation du modèle, soit en coefficients spectraux consécutifs . **Le rangement des coefficients spectraux est différent de celui utilisé dans le modèle (dans les programmes du logiciel FA, le nombre d'onde zonal est l'indice de la boucle interne alors que le nombre d'onde total est l'indice de la boucle externe).**

Il est d'usage que les champs d'altitude la pression sol (variables prognostiques aux niveaux modèle) et le géopotentiel de surface soient représentés sous forme de coefficients spectraux, tous les autres champs étant représentés sous forme de points de grille.
Il est fortement préconisé de ne compacter le géopotentiel de

surface ni en spectral, ni en points de grille, de façon à ne pas altérer la représentation de ce champ lorsqu'on passe d'un mode de représentation à l'autre.

Le logiciel d'interface ne gère pas de transformation spectral/points de grille ou vice-versa.

Dans le cas d'un fichier ARPEGE on a fait par ailleurs les **hypothèses** suivantes:

- la **grille** est celle de collocation, est **symétrique par rapport à l'équateur** , avec un nombre de points ainsi qu'un nombre d'onde zonal maximum par parallèle pouvant varier en longitude (de manière croissante, au sens large, d'un pôle vers l'équateur).

- le **rangement** des valeurs des champs points de grille est **contigu**, du (pseudo-)pôle Nord au (pseudo-)pôle Sud, les points étant rangés d'Ouest en Est à partir du (pseudo-) méridien zéro, les points de même latitude étant adjacents. (remarque: actuellement le logiciel n'utilise pas explicitement ce mode de rangement, mais exploite juste le fait que les données sont contigües)

Dans le cas d'un fichier ALADIN, les hypothèses homologues à ce qui précéde sont que:

- la grille est celle de collocation, avec un nombre fixe de points par "ligne".

- le rangement des valeurs des champs points de grille est contigu, du coin Sud-Ouest au coin Nord-Est, les points étant rangés consécutivement d'Ouest en Est.

En ce qui concerne les <u>vecteurs</u> :
- dans ARPEGE, les composantes "zonale" et "méridienne" sont relatives à la sphère transformée, c'est à dire que le repère local est celui de la sphère déformée et que le module des composantes n'est pas réel mais réduit ;
- dans ALADIN, les composantes "U" et "V" sont relatives au repère local du modèle, mais sont dépourvues du facteur d'échelle (car celui-ci n'est défini qu'à un facteur multiplicatif près) : au repère près, il s'agit donc de vecteurs "vus sur la grille réelle".

### 1.3) Auto-documentation

Tout fichier ARPEGE/ALADIN est **décrit, pour la partie documentaire significative**, par ce qu'on appelle par convention **un "cadre"**, et aussi par **une "date"**.

Un **"cadre"** est d'abord un ensemble de variables définies par un programme applicatif et stockées dans les variables globales du logiciel FA. On peut voir le cadre comme un objet informatique, pas nécessairement rattaché à un fichier. Le cadre fait aussi partie des fichiers FA et y est écrit sous la forme des **articles** suivants:

| Nom article (nom variable) | Dimension ARPEGE | Signification ARPEGE | Dimension Aladin | Signification Aladin |
|---|---|---|---|---|
| **CADRE-DIMENSIONS** (MTRONC) | 5 entiers | 1: NSMAX= troncature | 5 entiers | 1: NSMAX= troncature N/S |
| (NLATIT) | | 2: NDGL=nb de latitudes pôle à pôle | | 2: NDGL=nb de latitudes |
| (NXLOPA) | | 3: NDLON=nb max de longitudes par parallèle | | 3: NDLON=nb de longitudes |
| (NNIVER) | | 4: NFLEVG=nb de niveaux | | 4: NFLEVG=nb de niveaux |
| (NTYPTR) | | 5: NSTTYP=type de transformation horizontale | | 5: (-1)*NMSMAX= -troncature W/E |
| **CADRE-FRANKSCHMI** (SSLAPO) | 4 réels | 1: sinus de la latitude du pôle d'intérêt | 4 réels | 1: EBETA= angle de rotation "beta" du domaine |
| (SCLOPO) | | 2: cosinus de la longitude du pôle d'intérêt | | 2: inutilisé |
| (SSLOPO) | | 3: sinus de la longitude du pôle d'intérêt | | 3: inutilisé |
| (SCODIL) | | 4: coefficient de dilatation | | 4: PCODIL= 0. (cas réel, avec nécessité de définir une zone E) ou 1. (cas académique, domaine torique à la base) |

| | | | | |
|---|---|---|---|---|
| **CADRE-REDPOINPOL**<br>(NLOPAR) | 2*inpahe entiers, avec inpahe = (1+nb total de latitud.)/2 | 1->inpahe: nb de longitudes par parallèle de l'HN | 8 + 2*NSMAX + 4 entiers | KNLOPA(1): sous-troncature non compactée (à ne plus utiliser)<br>KNLOPA(2): =0 si le champ ne décrit que C+I en pts de grille, -1 s'il décrit C+I+E en pts de grille, 1 s'il décrit C+I+E en pts de grille et si les champs dynamiques sont en coeff. spect.<br>KNLOPA(3): KDLUN=1<br>KNLOPA(4): KDLUX= dim W/E domaine C+I<br>KNLOPA(5): KDGUN=1<br>KNLOPA(6): KDGUX= dim N/S domaine C+I<br>KNLOPA(7:8): KDZONL, KDZONG= nb de pts de la bande de couplage (largeur zonale et méridienne) |
| (NOZPAR) | | inpahe+1 -> 2*inpahe: nb d'onde zonal maximum par parallèle de l'HN | | non-utilisé |

| | | | | |
|---|---|---|---|---|
| **CADRE-SINLATITUD**<br>(SINLAT) | inpahe=(1+nb total de latitud)/2 réels | 1->inpahe: sinus des latitudes de la grille (Hémisphère Nord seulement) | 18 réels | 1: KROTEQ=0 (pas de rotation) ou 1 (pt(plonr,platr) basculé sur l'équateur)<br>2,3: plonr et platr, long. et lat. du pt de référence pour une éventuelle rotation (radian)<br>4,7: plon1, plat1, plon2, plat2, long. et lat. des coins SW et NE du domaine C+I (radian)<br>8,9: plon0, plat0, long. et lat. du pt référence pour la projection (radian)<br>10: PRPK= paramètre de projection<br>11: KSOTRP= paramètre d'isotropie pour l'appel à EGGX (sans objet ici)<br>12: KGIV0=type de choix du pt de référence pour la projection<br>13:,14: plx, ply, dim du domaine C+I (en m) dans les directions x et y<br>15,16: pdelx, pdely, taille de la maille (en m) dans les directions x et y<br>17,18: PXWN, PYWN |
| **CADRE-FOCOHYBRID**<br>(NPREFE)<br>(SFOHYB) | 1+2*(1+nb de niveaux) réels | pression de référence (Prefer) pour la fonction A, puis fonctions A/Prefer et B de la coordonnée verticale hybride (VALH etVBH) | 1+2*(1+nb de niveaux) réels | pression de référence (Prefer) pour la fonction A, puis fonctions A/Prefer et B de la coordonnée verticale hybride (VALH etVBH) |
| "Nom du cadre" associé au fichier | 1 entier | Numéro de version du logiciel FA (=1) | 1 entier | Numéro de version du logiciel FA (=1) |

Une **"date"** est, quant à elle, un tableau de 11 valeurs dans la syntaxe GRIB, à savoir dans l'ordre:

- année (civile, non limitée aux 2 derniers chiffres), mois, jour, heure, minute du réseau de départ;

- indicateur d'unité d'échéance (1 pour des heures, de préférence), suivi de 3 valeurs caractérisant cette échéance, qui doivent être en général:

0, 0, 0 pour une analyse (non initialisée);

N, 0, 10 pour une prévision à N heures d'échéance (N#0);

0, 0, 1 pour une analyse initialisée (prévi zéro heure).

- 2 valeurs qui n'ont de sens que pour les moyennes ou les cumuls temporels et qui doivent être nulles pour les cas cités ci-dessus.

**Depuis leur cycles respectif CY24T1 et AL15**, les programmes Arpege et Aladin initialisent le 10ème élément avec l'échéance du précédent fichier de sortie créé ; de sorte qu'il est possible, pour un fichier individuel, de retrouver la durée du cumul temporel.

**Le contenu du cadre et de la date sont écrits sur le fichier.**

**Pour un fichier en mode création, le cadre doit être défini avant l'ouverture .**

**Pour un fichier préexistant, ce n'est pas nécessaire** (le cadre est alors lu sur fichier lors de l'ouverture, de même que la date, et cela offre une possibilité de traitement dynamique), **mais si le cadre est prédéfini, il doit y avoir un accord strict entre celui-ci et celui lu sur fichier.**

Un cadre est repéré dans le logiciel par un nom symbolique attribué par le programme utilisateur. Ce nom peut très bien différer pour un même fichier utilisé dans 2 programmes distincts; le nom du cadre est une entité en mémoire; si on peut le retrouver sur fichier (cf. notion d'identificateur de fichier) ce n'est en aucun cas une contrainte lors de la réutilisation du fichier.

Un même cadre peut servir à plusieurs fichiers.

Un cadre ne pourra être redéfini (ou supprimé) que s'il n'est rattaché à aucun fichier ouvert, une telle redéfinition donnant lieu à un message d'avertissement et n'étant pas possible lors d'une ouverture.

La redéfinition d'une date pour un fichier ouvert et déjà muni d'une date est possible, mais donne lieu à un message d'avertissement.

Il existe aussi, bien qu'on puisse l'ignorer à l'utilisation, la notion d'**identificateur de fichier** : on peut attribuer, obtenir, modifier un nom (interne) à un fichier. A la création du fichier, le nom symbolique du cadre sert d'identificateur initial. Cet étiquetage peut servir à qualifier la nature du fichier (ex: type d'ébauche).

### 1.4) Options d'encodage

### A/ Description

**Un certain nombre d'options** (nécessaires pour écrire les champs, à cause du codage GRIB) **sont implicites, modifiables au niveau d'un fichier ouvert** , les **valeurs implicites** (utilisées lors de chaque ouverture) étant **elles-mêmes réglables**:

- le type de codage (pas de codage, GRIB standard ou GRIB amendé ARPEGE);

- le nombre de bits par valeur point de grille, et celui par coefficient spectral;

- la dimension de la "sous-troncature non compactée" (**la troncature est** supposée triangulaire dans ARPEGE et elliptique dans ALADIN);

- la "puissance de Dolby" P des champs en coefficients spectraux, ainsi que le "degré de modulation" D de celle-ci: afin de traiter ce type de champs, le compactage est appliqué sur un champ prétraité, ayant subi un opérateur (laplacien dans le cas ARPEGE, ad hoc dans le cas Aladin) à une puissance (entière) optimale au sens du critère d'erreur relative multipliée par l'erreur absolue due au compactage, en partant de P et dans la fouchette [P-D,P+D], en s'arrêtant au premier minimum relatif.

### B/ Valeurs implicites

Actuellement, les valeurs implicites *correspondantes* **des options d'encodage** sont les suivantes **mais** (sauf pour le niveau d'encodage GRIB) ce ne sont pas les valeurs utilisées dans la chaine opérationnelle :

- GRIB amendé ARPEGE (conservation exacte des extrema et des coefficients de la sous-troncature, utilisation optimale des degrés de liberté du compactage);

- 24 bits pour chaque valeur élémentaire compactée (tant point de grille que spectrale);

- T10 comme sous-troncature non compactée;

- 1 comme puissance de laplacien implicite, modulable dans une fourchette de plus ou moins 5 fois.

### C/ Valeurs de l'opérationnel

Les valeurs dites "par défaut", utilisées en opérationnel et qui sont générées en fournissant la valeur -1 aux arguments des sous-programmes FAGOTE ou FAGIOT sont :

- 16 bits pour chaque valeur élémentaire compactée en point de grille (24 bits avant le 23 février 2000);

- 16 (pour un fichier ARPEGE) ou 18 (pour un fichier ALADIN) bits pour chaque valeur élémentaire compactée en spectral (24 bits avant le 23 février 2000);

- sous-troncature variable selon une fonction des troncatures et du nombre de bits par champ spectral, de manière

différenciée dans le cas global (ARPEGE) et aire limitée (ALADIN). On prend alors le maximum des 3 termes suivants :
- Une fonction croissante de la troncature (linéaire affine, à l'arrondi près : (1+T)/10) (la plus grande des deux dans le cas ALADIN)
- Une fonction décroissante du nombre B de bits utilisés pour codés les coefficients spectraux (linéaire affine par rapport à l'inverse du nombre de bits, à l'arrondi près) qui dépend aussi de la troncature dans le cas ALADIN : ((1+T)*25)/(10*B), contre 480/B-10 pour ARPEGE
- L'ancienne valeur de réglage (=10, soit une sous-troncature fixe pour ARPEGE et Aladin jusqu'au 23 février 2000), avec une décote à (T-1) pour les faibles résolutions.

(l'application de la formule ci-dessus donne une sous-troncature de 20 pour ARPEGE T199 ou T149, de 13 pour ALADIN-France associé, et de 10 pour les autres ALADIN.

**Remarque** : l'application de cette formulation pour les champs spectraux peut amener, <u>pour les petites troncatures</u>, à ce qu'un champ compacté soit **plus long** que le même champ non compacté ... Dans la prochaine version du logiciel, on prévoira, dans un tel cas de figure, de supprimer tout compactage spectral.

# 2) GENERALITES SUR LES INTERFACES

## 2.1) Repérage des fichiers

Un fichier ARPEGE est repéré par un **numéro d'unité logique FORTRAN.** Il est possible de contacter le fichier par nom lors de son ouverture.

## 2.2) Repérage des champs

Les articles de données doivent être des **données consécutives en mémoire** : cette remarque est importante pour les champs.

Les **champs** sont, comme indiqué plus haut, **implicitement codés dans une forme de type GRIB.** Pour insérer d'autres types de données que les champs, seules des interfaces d'écriture et de lecture simple (sans aucune mise en forme ni compactage) sont disponibles. Ces dernières possibilités n'ont apparemment guère servi sauf en mémoire distribuée de manière couplée à des routines de codage/décodage, mais l'expérience des changements de calculateur conduit aux (vives) recommandations suivantes: ne former que des articles **homogènes** en type de données, et préfixer les noms d'articles associés de manière explicite: par exemple, par '%R%' pour des réels, '%I%' pour des entiers.

Les **champs** sont repérés par un **préfixe**, un **éventuel numéro de niveau** , et un **suffixe**; ces 3 entités définissent le nom d'article associé au champ. Le préfixe est soit le type de coordonnée verticale, auquel cas le niveau sert effectivement, soit conventionnel ; dans ce cas le niveau ne sert pas, et la distinction préfixe/suffixe est alors assez formelle. Le préfixe est limité à 8 caractères. Si le nom d'article généré dépasse les 16 caractères admissibles par LFI, il y a troncature du suffixe, ce qui ne pose aucun problème autre qu'un faible risque d'homonymie.
Les préfixes gérés dans le logiciel à ce jour (et qui n'empêchent pas d'en utiliser d'autres) sont les suivants :
- "S" : niveau hybride
- "P" : niveau isobare
- "H" : hauteur (au-dessus d'un relief de référence)
- "V" : niveau iso-tourbillon_potentiel
- "T" : niveau isentrope
- "SURF" : niveau de la surface
- "JET" : niveau du jet (au sens OACI)
- "TROPO" : niveau de la tropopause (au sens OACI)
- "MER" : niveau moyen de la mer

## 2.3) Environnement des interfaces

Les **noms des sous-programmes** d'interface sont **préfixés par "FA"**, comme " **F**ichiers **A**RPEGE".

Ces sous-programmes s'appuient sur le logiciel de fichiers indexés LFI, et sont écrits dans le même esprit. Ainsi y retrouve-t-on les mêmes notions en ce qui concerne la **messagerie**, le **traitement des erreurs**, le fonctionnement en mode multi-tâches ou en mode mise au point.

Par ailleurs, le fonctionnement interne du logiciel nécessite d'avoir des **limites de taille**, tant en ce qui concerne la **résolution horizontale et verticale** que le nombre de fichiers ouverts simultanément. Les limites effectives, configurables car ce sont des "PARAMETER" FORTRAN définis par des identificateurs de directives, sont les suivantes :

sur "CRAY", "SX4" et "VPP" :
- Troncature T511, 999 niveaux verticaux

sur "LOWRES", "DEC", "RS6K", "HPPA" et "SUN" :
- Troncature T215, 100 niveaux verticaux

Pour tous :
- 512 latitudes ; 1024 longitudes
- en recompilant les interfaces, la limite verticale peut être portée à 999 niveaux, sans affecter la relecture de fichiers écrits antérieurement
- 20 fichiers ouverts simultanément
- 20 cadres définis à la fois.

**Remarque importante pour éviter des débordements de tableaux:**

Lors de la **récupération du cadre d'un fichier que l'on a ouvert sans cadre prédéfini** , on récupère par un même sous-programme (FACIES) **les variables simples et les tableaux.** Même si un programme utilisateur est écrit pour lire n'importe quel fichier ARPEGE, il a forcément des limites sur les tailles des tableaux, donc en termes de résolutions. Si ces limites sont inférieures à celles du logiciel d'interface, il peut donc y avoir débordement de tableaux lors de la récupération du cadre si la résolution horizontale ou verticale "du fichier" dépasse les capacités du programme utilisateur et pas celles du logiciel.

**Pour se prémunir d'un tel cas de figure, il est possible -et chaudement recommandé- de spécifier des limites utilisateur en termes de résolution (FARFLU), ou éventuellement de s'aligner au préalable sur les limites du logiciel (FALIMU, avant tout appel à FARFLU).**

# 3) DESCRIPTION DES INTERFACES UTILISATEUR

## 3.1) Liste et fonctionnalités

Les sous-programmes d'interface utilisateur réalisés début 2000 assurent les fonctionnalités suivantes (un sous-programme par fonctionnalité):

| sous-programme | catégorie | Fonctionnalité |
|---|---|---|
| FACADE | basique | Définition d'un cadre |
| FAITOU | basique | Ouverture d'un fichier, qui peut être contacté par nom |
| FANDAR | basique | Définition d'une date associée à un fichier ouvert |
| FAIENC | basique | Ecriture d'un champ sur un fichier |
| FACILE | basique | Lecture d'un champ sur un fichier |
| FACIES | basique | Obtention du contenu d'un cadre |
| FADIES | basique | Obtention du contenu d'une date |
| FANION | basique | Obtention des caractéristiques d'un article de type "champ horizontal" |
| FAIRME | basique | Fermeture d'un fichier |
| FACOND | mémoire distribuée | Codage d'un champ destiné à être écrit sur un fichier |
| FADECO | mémoire distribuée | Décodage d'un champ venant d'être lu sur un fichier |
| FAIRNO | mémoire distribuée | Fermeture non explicite d'une unité logique d'un fichier (pour processeurs n'effectuant pas d'I/O) |
| FANOUV | mémoire distribuée | Initialisations (sans ouverture explicite) de l'environnement d'un fichier (pour processeurs n'effectuant pas d'I/O) |
| FAGOTE | réglage | Modification des options de codage GRIB propres à un fichier |
| FAGIOT | réglage | Modification des options implicites de codage GRIB |
| FAVEUR | réglage | Obtention des options de codage GRIB propres à un fichier |
| FAVORI | réglage | Obtention des options implicites de codage GRIB |
| FACTUM | gestion | Suppression d'un cadre |
| FACAGE | gestion | Modification de l'option de conservation d'un cadre |
| FAISAN | avancé | Ecriture d'un article "simple" (c'est-à-dire non assimilable à un champ horizontal) sur un fichier. (Couplage possible avec FACOND en mémoire distribuée) |
| FALAIS | avancé | Lecture d'un article "simple" (couplage possible avec FADECO en mémoire distribuée). |
| FARINE | avancé | Initialisation du logiciel FA |
| FAMISO | préférences | Fonctionnement mise au point ou non |
| FANMSG | préférences | Modulation de la messagerie au niveau global |
| FANERG | préférences | Modulation du traitement des erreurs au niveau global |

| | | |
|---|---|---|
| FANIME | préférences | Modulation de la messagerie pour un fichier |
| FATALE | préférences | Mode de traitement des erreurs pour un fichier |
| FARFLU | logiciel | Réglage de limites usager en termes de résolutions horizontale et verticale |
| FALIMU | logiciel | Obtention des limites usager (par défaut, limites logiciel) |
| FACOCH | autres | Copie d'un article de type "champ" d'un fichier sur un autre, "en transparent", sans décodage/recodage |
| FAUTIF | autres | Définition de l'identificateur d'un fichier |
| FALSIF | autres | Obtention de l'identificateur d'un fichier |

### 3.2) Recommandations

**Dans le cas d'un programme écrivain d'un nouveau fichier** , il faut avant de pouvoir écrire des données sur ce fichier:

- Définir le cadre s'il n'existe pas encore (en notant qu'un même cadre peut servir à plusieurs fichiers) (FACADE);

- Ouvrir le fichier (FAITOU);

- Définir la date (FANDAR);

... et **ne pas oublier de fermer le fichier avant de terminer le programme**, pour pouvoir le réutiliser ultérieurement (FAIRME).

**Dans le cas d'un programme utilisant un fichier préexistant** , il faut avant de pouvoir lire les données du fichier:

- **Si on s'attend à traiter une configuration particulière** et uniquement celle-ci, **définir le cadre** s'il n'existe pas encore (FACADE);

- **Si au contraire on est prêt à traiter un fichier de configuration "quelconque", préciser au préalable les limites acceptables en termes de résolution horizontale et verticale** (FARFLU);

- Ouvrir le fichier, en référençant éventuellement un cadre préexistant (ce qui impose un accord strict entre ce cadre et celui présent sur le fichier) (FAITOU).

Il est ensuite recommandé de **demander la date** (FADIES), et de **la contrôler si besoin est** (il y a un contrôle syntaxique de celle-ci à la définition). Par ailleurs on recommande de **fermer le fichier quand on n'en a plus besoin** , au plus tard avant la fin du programme.

**Une remarque concernant les cadres:**

Lorsqu'un utilisateur définit explicitement un cadre (par FACADE), il précise dans les arguments d'appel -outre un nom symbolique et les caractéristiques numériques- si l'on devra garder ou pas (en mémoire, au niveau du logiciel d'interface) le cadre lors de la fermeture du dernier fichier s'y rattachant.

Dans le cas d'un **cadre qui a été défini dynamiquement**, c'est-à-dire **créé par l'ouverture d'un fichier préexistant**, il y a **implicitement suppression de ce cadre lors de la fermeture du (dernier) fichier s'y rattachant.**

Si l'on souhaite éviter ceci, il faut spécifier via FACAGE que l'on veut garder ledit cadre, avant ladite fermeture.

## Recopie de champs:

Pour recopier sur un fichier résultat les **champs de "constantes" ou qui n'évoluent pas**, par exemple **l'indicateur terre/mer, le relief**, on recommande fortement de **recopier les articles correspondants à l'aide du sous-programme "FACOCH" .** Ainsi évite-t-on le recodage du champ, avec modification potentielle de celui-ci (via des options GRIB différentes, en particulier si le champ d'origine n'a pas les options implicites et/ou en cas de changement de calculateur)... bref **on évite la variation des constantes.**

## Au sujet des très vieux fichiers:

Les éventuels fichiers ARPEGE créés avec la toute première version du logiciel d'interface (cycle 2 de la bibliothèque auxiliaire) ne sont pas lisibles avec les versions ultérieures, à cause de l'ajout du type de transformation horizontale dans le cadre, et à cause de l'ajout d'un article dont le nom est l'identificateur de fichier,article qui doit avoir un emplacement précis dans le fichier.

Si l'on veut néanmoins récupérer des articles, même de type "champ horizontal", il est possible de manipuler de tels fichiers directement via les interfaces du logiciel LFI (la liste des noms d'articles peut être obtenue à l'aide du sous-programme LFILAF).

# 4) ENTETES DES INTERFACES UTILISATEUR

## facade

```
      SUBROUTINE FACADE ( CDNOMC, KTYPTR, PSLAPO, PCLOPO, PSLOPO,
     S                          PCODIL, KTRONC, KNLATI, KNXLON, KNLOPA,
     S                          KNOZPA, PSINLA, KNIVER, PREFER, PAHYBR,
     S                          PBHYBR, LDGARD )
C****
C Sous-programme servant a DEfinir un CADre, voire a le
C redefinir.
C**
C      Arguments : CDNOMC == Nom symbolique du cadre;
C (tous d'Entree)  KTYPTR == Type de transformation horizontale;
C                  PSLAPO == Sinus de la latitude du pole d'interet;
C                  PCLOPO == Cosinus " " longitude " " " ;
C                  PSLOPO == Sinus " " longitude " " " ;
C                  PCODIL == Coefficient de dilatation;
C                  KTRONC == Troncature;
C                  KNLATI == Nombre de latitudes (de pole a pole);
C                  KNXLON == Nombre maxi de longitudes par parallele;
C        (Tableau) KNLOPA == Nombre de longitudes par parallele;
C                            (du pole nord vers l'equateur seulement)
C        (Tableau) KNOZPA == Nombre d'onde zonal maxi par parallele;
C                            (du pole nord vers l'equateur seulement)
C        (Tableau) PSINLA == Sinus des latitudes de l'hemisphere nord
C                            (du pole nord vers l'equateur seulement)
C                  KNIVER == Nombre de niveaux verticaux;
C                  PREFER == Pression de reference (facteur multipli-
C                            catif de la premiere fonction de la
C                            coordonnee hybride)
C        (Tableau) PAHYBR == Valeurs de la fonction "A" de la coordo-
C                            nnee hybride AUX LIMITES DE COUCHES;
C        (Tableau) PBHYBR == Valeurs de la fonction "B" de la coordo-
C                            nnee hybride AUX LIMITES DE COUCHES;
C                  LDGARD == Vrai si le cadre doit etre conserve meme
C                            apres la fermeture du dernier fichier
C                            qui s'y rattache.
C*
C La "redefinition" d'un cadre est possible a l'une de ces
C conditions:
C
C - le cadre a ete defini, mais n'a aucun fichier qui s'y rattache;
C - le cadre defini a au moins un fichier qui s'y rattache, et les
C nouveaux parametres de definition sont identiques a ceux deja
C definis.
C
C Toute "redefinition" de cadre donne lieu a une messagerie
C de niveau 1, donc non masquee par defaut.
C
```

## facage

```
      SUBROUTINE FACAGE ( CDNOMC, LDGARD )
C****
C Sous-programme servant a redefinir l'option de conservation
C d'un cadre preexistant ( CAdre a Garder Eventuellement... )
C**
C Arguments :     CDNOMC == Nom symbolique du cadre;
C (tous d'Entree) LDGARD == Vrai si le cadre doit etre conserve meme
C                           apres la fermeture du dernier fichier
C                           qui s'y rattache.
C
```

### facies

```
      SUBROUTINE FACIES ( CDNOMC, KTYPTR, PSLAPO, PCLOPO, PSLOPO,
     S                          PCODIL, KTRONC, KNLATI, KNXLON, KNLOPA,
     S                          KNOZPA, PSINLA, KNIVER, PREFER, PAHYBR,
     S                          PBHYBR, LDGARD )
C****
C Sous-programme servant a obtenir le contenu d'un Cadre.
C ( FACIES... par analogie avec FADIES, avec "C" pour Cadre... )
C**
C Arguments :      CDNOMC == Nom symbolique du cadre;
C (tous de Sortie, KTYPTR == Type de transformation horizontale
C sauf CDNOMC)     PSLAPO == Sinus de la latitude du pole d'interet;
C                  PCLOPO == Cosinus " " longitude " " " ;
C                  PSLOPO == Sinus " " longitude " " " ;
C                  PCODIL == Coefficient de dilatation;
C                  KTRONC == Troncature;
C                  KNLATI == Nombre de latitudes (de pole a pole);
C                  KNXLON == Nombre maxi de longitudes par parallele;
C        (Tableau) KNLOPA == Nombre de longitudes par parallele;
C                            (du pole nord vers l'equateur seulement)
C        (Tableau) KNOZPA == Nombre d'onde zonal maxi par parallele;
C                            (du pole nord vers l'equateur seulement)
C        (Tableau) PSINLA == Sinus des latitudes de l'hemisphere nord
C                            (du pole nord vers l'equateur seulement)
C                  KNIVER == Nombre de niveaux verticaux;
C                  PREFER == Pression de reference (facteur multipli-
C                            catif de la premiere fonction de la
C                            coordonnee hybride)
C        (Tableau) PAHYBR == Valeurs de la fonction "A" de la coordo-
C                            nnee hybride AUX LIMITES DE COUCHES;
C        (Tableau) PBHYBR == Valeurs de la fonction "B" de la coordo-
C                            nnee hybride AUX LIMITES DE COUCHES;
C                  LDGARD == Vrai si le cadre doit etre conserve meme
C                            apres la fermeture du dernier fichier
C                            qui s'y rattache.
C
```

### facile

```
       SUBROUTINE FACILE ( KREP, KNUMER, CDPREF, KNIVAU, CDSUFF, KCHAMP,
     S                          LDCOSP )
C****
C Sous-programme de LECTURE d'un CHAMP HORIZONTAL sur un fichier
C ARPEGE.
C ( Champ d'Interet en LEcture )
C**
C Arguments :   KREP (Sortie) == Code-reponse du sous-programme;
C             KNUMER (Entree) == Numero de l'unite logique;
C             CDPREF (Entree) == Prefixe eventuel du nom d'article;
C             KNIVAU (Entree) == Niveau vertical eventuel;
C             CDSUFF (Entree) == Suffixe eventuel du nom d'article;
C ( Tableau ) KCHAMP (Sortie) == Valeurs REELLES du champ lu;
C             LDCOSP (Entree) == Vrai si le champ est represente
C                                par des coefficients spectraux.
```

### facoch

```
       SUBROUTINE FACOCH ( KREP, KNUME1, KNUME2, CDPREF, KNIVAU, CDSUFF )
C****
C Sous-programme de reCOpie d'un Champ Horizontal d'un fichier
C ARPEGE sur un autre .
C**
C Arguments : KREP (Sortie) == Code-reponse du sous-programme;
C           KNUME1 (Entree) == Numero d'unite logique en entree;
C           KNUME2 (Entree) == Numero d'unite logique en sortie;
C           CDPREF (Entree) == Prefixe eventuel du nom d'article;
C           KNIVAU (Entree) == Niveau vertical eventuel;
C           CDSUFF (Entree) == Suffixe eventuel du nom d'article.
C
```

### facond

```
      SUBROUTINE FACOND ( KREP, KNUMER, CDPREF, KNIVAU, CDSUFF,
     S                         PCHAMP, LDCOSP, CDNOMA, KLNOMA, PVALCO,
     S                         KLONGD )
#include "facom0.h"
C****
C Sous-programme de CODAGE d'un CHAMP HORIZONTAL destine a etre
C sur un fichier ARPEGE/ALADIN.
C ( COdage de (Nouvelles ?) Donnees )
C**
C Arguments :   KREP (Sortie) == Code-reponse du sous-programme;
C             KNUMER (Entree) == Numero de l'unite logique;
C             CDPREF (Entree) == Prefixe eventuel du nom d'article;
C             KNIVAU (Entree) == Niveau vertical eventuel;
C             CDSUFF (Entree) == Suffixe eventuel du nom d'article;
C ( Tableau ) PCHAMP (Entree) == Valeurs REELLES du champ a ecrire;
C             LDCOSP (Entree) == Vrai si le champ est represente
C                                 par des coefficients spectraux;
C             CDNOMA (Sortie) == Nom de l'article-champ a ecrire;
C             KLNOMA (Sortie) == Nombre de caracteres utiles dans
C                                 CDNOMA;
C ( Tableau ) PVALCO (Sortie) == Donnees destinees a l'ecriture;
C             KLONGD (Sortie) == Nombre de valeurs (mots de 64 bits
C                                 en principe) a ecrire.
C
C Remarques:
C
C - PVALCO est type reel par commodite, et doit avoir une longueur
C suffisante pour stocker les donnees codees. Le dimensionnement
C "tous terrains" est (2+JPXCHA), qui permet le cas echeant de
C stocker un champ a pleine resolution sans codage effectif.
C
C - CDNOMA doit avoir au moins JPXNOM caracteres.
```

### factum

```
      SUBROUTINE FACTUM ( CDNOMC )
C****
C Sous-programme servant a supprimer un cadre.
C ( Cadre a TUer Methodiquement ? )
C**
C Argument : CDNOMC (Entree) == Nom symbolique du cadre.
C
```

### fadeco

```
      SUBROUTINE FADECO ( KREP, KNUMER, CDPREF, KNIVAU, CDSUFF,
     S                          LDCOSP, CDNOMA, KLNOMA, PVALCO, KLONGD,
     S                          PCHAMP )
C Sous-programme de controle et de DECODAGE d'un CHAMP HORIZONTAL
C venant d'etre lu sur un fichier ARPEGE/ALADIN.
C ( DECOdage de donnees )
C**
C Arguments :   KREP (Sortie) == Code-reponse du sous-programme;
C             KNUMER (Entree) == Numero de l'unite logique;
C             CDPREF (Entree) == Prefixe eventuel du nom d'article;
C             KNIVAU (Entree) == Niveau vertical eventuel;
C             CDSUFF (Entree) == Suffixe eventuel du nom d'article;
C             LDCOSP (Entree) == Vrai si le champ est represente
C                                 par des coefficients spectraux;
C             CDNOMA (Sortie) == Nom de l'article-champ lu;
C             KLNOMA (Sortie) == Nombre de caracteres utiles dans
C                                 CDNOMA;
C ( Tableau ) PVALCO (Entree) == Donnees issues de la lecture;
C             KLONGD (Entree) == Nombre de valeurs (mots de 64 bits
C                                 en principe) lues;
C ( Tableau ) PCHAMP (Sortie) == Valeurs REELLES du champ lu.
```

### fadies

```
      SUBROUTINE FADIES ( KREP, KNUMER, KDATEF )
C****
C Sous-programme permettant d' obtenir la date d'un fichier ouvert
C pour le logiciel de Fichiers ARPEGE, et deja muni d'une date.
C ( "DIES" = jour en latin... )
C**
C Arguments : KREP (Sortie) == Code-reponse du sous-programme;
C           KNUMER (Entree) == Numero de l'unite logique;
C (Tableau) KDATEF (Sortie) == Date elle-meme (JPLDAT mots).
C
```

## fagiot

```
      SUBROUTINE FAGIOT ( KNGRIB, KNBPDG, KNBCSP, KSTRON, KPUILA,
     S                          KDMOPL )
C****
C Ce sous-programme permet de modifier les options implicites
C liees au codage GRIB des champs .
C CES OPTIONS NE SONT UTILISEES QUE POUR (RE)ECRIRE DES CHAMPS
C codes en GRIB, et les nouvelles valeurs implicites ne serviront
C que LORS d'une OUVERTURE de FICHIER ULTERIEURE.
C ( Grib, Implicites Options Techniques )
C**
C Arguments : KNGRIB (Entree) == Niveau de codage GRIB (0,1,2);
C             KNBPDG (Entree) == Nombre de bits par valeur point-
C                                 de-grille;
C             KNBCSP (Entree) == Nombre de bits par partie reelle/
C                                 imaginaire de coeff. spectral;
C             KSTRON (Entree) == Sous-troncature non compactee;
C             KPUILA (Entree) == Puissance de laplacien;
C             KDMOPL (Entree) == Degre de modulation de KPUILA.
C
C N.B.: Il doit y avoir coherence vis-a-vis des cadres deja definis
C et vis-a-vis des limites usagers.
C ( ce qui en pratique, ne concerne que KSTRON )
C
```

## fagote

```
      SUBROUTINE FAGOTE ( KREP, KNUMER, KNGRIB, KNBPDG, KNBCSP, KSTRON,
     S                          KPUILA, KDMOPL )
C****
C Ce sous-programme permet d'ajuster, pour un fichier ARPEGE
C ouvert, les options liees au codage GRIB des champs.
C CES OPTIONS NE SONT UTILISEES QUE POUR (RE)ECRIRE DES CHAMPS
C codes en GRIB.
C ( Grib, Options Techniques Effectives )
C**
C Arguments : KREP (Sortie) == Code-reponse du sous-programme;
C             KNUMER (Entree) == Numero d'Unite Logique concernee;
C             KNGRIB (Entree) == Niveau de codage GRIB (0,1,2);
C             KNBPDG (Entree) == Nombre de bits par valeur point-
C                                 de-grille;
C             KNBCSP (Entree) == Nombre de bits par partie reelle/
C                                 imaginaire de coeff. spectral;
C             KSTRON (Entree) == Sous-troncature non compactee;
C             KPUILA (Entree) == Puissance de laplacien;
C             KDMOPL (Entree) == Degre de modulation de KPUILA.
C
```

### faienc

```
      SUBROUTINE FAIENC ( KREP, KNUMER, CDPREF, KNIVAU, CDSUFF, KCHAMP,
     S                           LDCOSP )
C****
C Sous-programme d'ECRITURE d'un CHAMP HORIZONTAL sur un fichier
C ARPEGE.
C ( Integration par Ecriture d'un (Nouveau ?) Champ )
C**
C Arguments :   KREP (Sortie) == Code-reponse du sous-programme;
C             KNUMER (Entree) == Numero de l'unite logique;
C             CDPREF (Entree) == Prefixe eventuel du nom d'article;
C             KNIVAU (Entree) == Niveau vertical eventuel;
C             CDSUFF (Entree) == Suffixe eventuel du nom d'article;
C ( Tableau ) KCHAMP (Entree) == Valeurs REELLES du champ a ecrire;
C             LDCOSP (Entree) == Vrai si le champ est represente
C                                 par des coefficients spectraux.
C
```

### fairme

```
      SUBROUTINE FAIRME ( KREP, KNUMER, CDSTTU )
C****
C Sous-programme de FERMETURE d'une unite logique "Fichier ARPEGE"
C**
C Arguments : KREP (Sortie) == Code-reponse du sous-programme;
C           KNUMER (Entree) == Numero de l'unite logique;
C           CDSTTU (Entree) == "STATUS" eventuel pour "CLOSE".
```

### fairno

```
      SUBROUTINE FAIRNO ( KREP, KNUMER, CDSTTU )
#include "facom0.h"
C****
C Sous-programme de FERMETURE d'une unite logique "Fichier ARPEGE"
C (utilise par des processeurs qui ne font pas d'ecriture effective
C sur disque)
C**
C Arguments : KREP (Sortie) == Code-reponse du sous-programme;
C           KNUMER (Entree) == Numero de l'unite logique;
C           CDSTTU (Entree) == "STATUS" eventuel pour "CLOSE".
C
```

### faisan

```
      SUBROUTINE FAISAN ( KREP, KNUMER, CDNOMA, PDONNE, KLONGD )
C****
C Sous-programme d' ecriture d'un article de donnees non assimila-
C bles a un champ horizontal sur un fichier ARPEGE.
C ( Integration Simple d'un Article Non code )
C**
C Arguments :   KREP (Sortie) == Code-reponse du sous-programme;
C             KNUMER (Entree) == Numero de l'unite logique;
C             CDNOMA (Entree) == Nom de l'article;
C ( Tableau ) PDONNE (Entree) == Donnees a ecrire;
C             KLONGD (Entree) == Nombre de mots a ecrire.
C
```

### faitou

```
      SUBROUTINE FAITOU ( KREP, KNUMER, LDNOMM, CDNOMF, CDSTTU, LDERFA,
     S                          LDIMST, KNIMES, KNBARP, KNBARI, CDNOMC )
C****
C Sous-programme d' OUVERTURE d'une unite logique "Fichier ARPEGE"
C Il s'agit d'un fichier indexe, traite par le logiciel LFI.
C
C**
C ARGUMENTS : Ce sont les memes que pour "LFIOUV", avec CDNOMC comme
C             argument supplementaire.
C
C          KREP (Sortie) == Code-reponse du sous-programme;
C        KNUMER (Entree) == Numero de l'unite logique;
C        LDNOMM (Entree) == Vrai si l'unite logique doit etre
C                            associee a un NOM de Fichier EXP-
C                             LICITE lors de l'"OPEN" FORTRAN;
C        CDNOMF (Entree) == Nom de fichier explicite, si
C                            *LDNOMM* est VRAI - Meme si ce
C                            n'est pas le cas, ce *DOIT* ETRE
C                            UN OBJET DE TYPE "CHARACTER" .
C        CDSTTU (Entree) == "STATUS" pour l'"OPEN" FORTRAN
C                            ('OLD','NEW','UNKNOWN','SCRATCH')
C                            par defaut, mettre 'UNKNOWN';
C        LDERFA (Entree) == Option d'erreur fatale;
C        LDIMST (Entree) == Option impression de Statistiques
C                            au moment de la fermeture;
C        KNIMES (Entree) == Niveau de la Messagerie (0,1 ou 2)
C                            ( 0==Rien, 2==Tout )
C        KNBARP (Entree) == Nombre d'articles logiques prevus,
C                            ce qui n'est utilise que lors de
C                            la Creation du fichier,
C                            et qui n'empeche quand meme pas
C                            d'avoir plus d'articles logiques;
C        KNBARI (Sortie) == Nombre d'articles logiques de don-
C                            nees sur le fichier, initialement.
C                            (zero si creation)
C        CDNOMC (Entree) == Nom du CADRE associe au fichier.
C*
C N.B. : Pour un fichier en mode creation, ce cadre doit avoir ete
C defini au prealable (via le sous-programme FACADE, ou par
C l'ouverture d'un fichier preexistant).
C Pour un fichier ARPEGE preexistant, le cadre est lu sur le
C fichier; s'il etait deja defini auparavant, il y a controle
C de coherence entre les deux versions du cadre.
C
```

### falais

```
      SUBROUTINE FALAIS ( KREP, KNUMER, CDNOMA, PDONNE, KLONGD)
C****
C Sous-programme de lecture d'un article de donnees non assimila-
C bles a un champ horizontal sur un fichier ARPEGE.
C ( Lecture d'un Article Integre Simplement, non code )
C**
C Arguments :   KREP (Sortie) == Code-reponse du sous-programme;
C             KNUMER (Entree) == Numero de l'unite logique;
C             CDNOMA (Entree) == Nom de l'article;
C ( Tableau ) PDONNE (Entree) == Donnees a ecrire;
C             KLONGD (Entree) == Nombre de mots a ecrire.
C
```

## falimu

```
      SUBROUTINE FALIMU ( KXNIVV, KXTRON, KXLATI, KXLONG )
C****
C Sous-programme servant a obtenir les valeurs courantes
C des LIMites Utilisateur en termes de Resolutions horizontale
C et verticale, valides globalement pour tous les Fichiers et Cadres
C ARPEGE manipules par le programme utilisateur.
C**
C Arguments :      KXNIVV == Nombre maximum de niveaux verticaux;
C (tous de Sortie) KXTRON == Troncature maximum;
C                  KXLATI == Nombre maximum de latitudes pole a pole;
C                  KXLONG == Nombre maxi de longitudes par parallele.
C
```

## falsif

```
      SUBROUTINE FALSIF ( KREP, KNUMER, CDIDEN )
C****
C Sous-programme renvoyant le NOM de l'Identificateur
C d'un fichier ARPEGE .
C ( Lecture Specifique de l'Identificateur de Fichier )
C**
C Arguments : KREP (Sortie) == Code-reponse du sous-programme;
C           KNUMER (Entree) == Numero de l'unite logique;
C           CDIDEN (Sortie) == Nom de l'identificateur.
C
```

## famiso

```
      SUBROUTINE FAMISO ( LDEBUG )
C****
C Ce sous-programme permet d'activer ou de desactiver le mode
C "Mise au point du logiciel" . ( par defaut, inactif )
C A noter que le mode "mise au point" du logiciel LFI n'est pas
C modifie.
C**
C Argument : LDEBUG (Entree) == Vrai si on doit activer ce mode.
C
```

## fandar

```
      SUBROUTINE FANDAR ( KREP, KNUMER, KDATEF )
C****
C Sous-programme de definition d'une (Nouvelle) Date sur un fichier
C ARpege.
C**
C Arguments : KREP (Sortie) == Code-reponse du sous-programme;
C           KNUMER (Entree) == Numero de l'unite logique;
C (Tableau) KDATEF (Entree) == Date elle-meme (JPLDAT mots).
C*
C En cas de modification effective (si le fichier etait deja muni
C d'une date), il y a messagerie de niveau 1.
C
```

## fanerg

```
      SUBROUTINE FANERG ( KNIVAU )
C****
C Ce sous-programme se charge de mettre le Niveau Global d'Erreur
C Fatale du logiciel de Fichiers ARPEGE (*NRFAGA*) a la valeur
C KNIVAU, de meme que la variable correspondante du logiciel LFI.
C Par defaut, NRFAGA vaut 1.
C**
C Argument : KNIVAU (Entree) == Niveau global d'erreur fatale.
C                               Valeurs possibles:
C
C 0 : Rendre fatale toute erreur detectee, meme si elle correspond
C     a un fichier ouvert avec l'option "pas d'erreur fatale".
C 1 : Ne rend fatales que certaines erreurs "globales", c'est-a-dire
C     non reliables a un fichier ouvert, et les erreurs sur les fi-
C     chiers ouverts avec l'option "erreur fatale" (Mode par defaut)
C 2 : Passer outre toute erreur detectee, meme si elle correspond
C     a un fichier ouvert avec l'option "erreur fatale".
C     Neanmoins, le code-reponse eventuel ne sera pas zero.
C
```

## fanime

```
    SUBROUTINE FANIME ( KREP, KNUMER, KNIMES )
C****
C Ce sous-programme permet d'ajuster le Niveau de Messagerie
C propre aux actions faites sur un fichier particulier, ouvert pour
C le Logiciel de Fichiers ARPEGE, de meme que le Niveau correspon-
C dant du logiciel LFI .
C Cependant, tant que le Niveau de Messagerie Global *NIMSGA*
C vaut 0 ou 2, le niveau propre au fichier est inoperant.
C *NIMSGA* vaut par defaut 1, et est reglable via le s/p "FANMSG".
C**
C Arguments : KREP (Sortie) == Code-reponse du sous-programme;
C           KNUMER (Entree) == Numero d'Unite Logique concernee;
C           KNIMES (Entree) == Niveau de Messagerie souhaite.
C
```

## fanion

```
      SUBROUTINE FANION ( KREP, KNUMER, CDPREF, KNIVAU, CDSUFF, LDEXIS,
     S                         LDCOSP, KNGRIB, KNBITS, KSTRON, KPUILA )
C****
C Sous-programme renseignant sur l'EXISTENCE et les CARACTERISTI-
C QUES eventuelles d'un Article de type CHAMP dans un Fichier ARPEGE
C ( LDEXIS est le "fanion" leve si l'article existe )
C**
C Arguments : KREP (Sortie) == Code-reponse du sous-programme;
C           KNUMER (Entree) == Numero de l'unite logique;
C           CDPREF (Entree) == Prefixe eventuel du nom d'article;
C           KNIVAU (Entree) == Niveau vertical eventuel;
C           CDSUFF (Entree) == Suffixe eventuel du nom d'article;
C           LDEXIS (Sortie) == Vrai si l'article de type CHAMP
C                               existe bien dans le Fichier;
C           LDCOSP (Sortie) == Vrai si le champ est represente
C                               par des coefficients spectraux;
C           KNGRIB (Sortie) == Niveau de codage GRIB;
C           KNBITS (Sortie) == Nombre de bits de codage eventuel;
C           KSTRON (Sortie) == Sous-troncature non codee " -le;
C           KPUILA (Sortie) == Puissance de laplacien eventuelle.
C
C KNBITS n'a de sens que si l'article existe et a ete code;
C de meme pour KSTRON et KPUILA, qui ne sont applicables qu'a un
C champ represente en coefficients spectraux.
C Les arguments de sortie n'ayant pas de sens sont mis a
C 0 pour les entiers, .FALSE. pour les logiques.
```

## fanmsg

```
      SUBROUTINE FANMSG ( KNIVAU )
C****
C Ce sous-programme se charge de mettre le Niveau Global d'
C impression des Messages du logiciel de Fichiers ARPEGE (*NIMSGA*)
C a la valeur KNIVAU , de meme que la variable correspondante du
C du logiciel LFI . Par defaut, NIMSGA vaut 1.
C**
C Argument : KNIVAU (Entree) == Niveau Global d'Impression
C                                des Messages.
C Valeurs possibles:
C
C 0 : N'emettre que les messages d'erreurs reellement importants .
C 1 : N'emettre qu'un minimum de messages "globaux", et les messages
C     lies a un fichier ouvert qui sont de niveau au plus egal au
C     niveau de la messagerie pour ce fichier (Mode par defaut) .
```

## fanouv

```
      SUBROUTINE FANOUV ( KREP, KNUMER, LDNOMM, CDNOMF, CDSTTU, LDERFA,
     S                         LDIMST, KNIMES, KNBARP, KNBARI, CDNOMC )
#include "facom0.h"
C****
C Sous-programme d'initialisation SANS OUVERTURE d'une unite logique
C "Fichier ARPEGE". Il s'agit d'un fichier indexe,
C traite par le logiciel LFI.
C
C FANOUV est derive de FAITOU, mais n'appelle pas la couche LFI
C pour l'ouverture reelle.
C
C utilise pour la seule compression des donnees par des processeurs
C qui ne font pas d'ecriture effective sur disque
C
C
C**
C ARGUMENTS : Ce sont les memes que pour "LFIOUV", avec CDNOMC comme
C argument supplementaire.
C
C       KREP (Sortie) == Code-reponse du sous-programme;
C     KNUMER (Entree) == Numero de l'unite logique;
C     LDNOMM (Entree) == Vrai si l'unite logique doit etre
C                        associee a un NOM de Fichier EXP-
C                        LICITE lors de l'"OPEN" FORTRAN;
C     CDNOMF (Entree) == Nom de fichier explicite, si
C                        *LDNOMM* est VRAI - Meme si ce
C                        n'est pas le cas, ce *DOIT* ETRE
C                        UN OBJET DE TYPE "CHARACTER" .
C     CDSTTU (Entree) == "STATUS" pour l'"OPEN" FORTRAN
C                        ('OLD','NEW','UNKNOWN','SCRATCH')
C                        par defaut, mettre 'UNKNOWN';
C     LDERFA (Entree) == Option d'erreur fatale;
C     LDIMST (Entree) == Option impression de Statistiques
C                        au moment de la fermeture;
C     KNIMES (Entree) == Niveau de la Messagerie (0,1 ou 2)
C                        ( 0==Rien, 2==Tout )
C     KNBARP (Entree) == Nombre d'articles logiques prevus,
C                        ce qui n'est utilise que lors de
C                        la Creation du fichier,
C                        et qui n'empeche quand meme pas
C                        d'avoir plus d'articles logiques;
C     KNBARI (Sortie) == Nombre d'articles logiques de don-
C                        nees sur le fichier, initialement.
C                        (zero si creation)
C     CDNOMC (Entree) == Nom du CADRE associe au fichier.
C*
C N.B. : Pour un fichier en mode creation, ce cadre doit avoir ete
C defini au prealable (via le sous-programme FACADE, ou par
C l'ouverture d'un fichier preexistant).
C Pour un fichier ARPEGE preexistant, le cadre est lu sur le
C fichier; s'il etait deja defini auparavant, il y a controle

C de coherence entre les deux versions du cadre.
```

## farine

```
      SUBROUTINE FARINE ( KOPTIO )
C****
C Ce sous-programme est charge des INITIALISATIONS du logiciel
C de Fichiers ARPEGE FA ( Routine d'INitialisation )
C**
C Argument :     KOPTIO == OPTION concernant le mode d'utilisation.
C (Entree)                 (MULTI-TACHE ou NON)
C VALEURS POSSIBLES : 0 == Mode MONO-Tache prescrit;
C                     1 == Mode MULTI-Taches prescrit;
C                     2 == Utilisation du mode par defaut si c'est
C                          le premier appel; sinon on garde le mode
C                          prescrit anterieurement .
C
```

## farflu

```
      SUBROUTINE FARFLU ( KXNIVV, KXTRON, KXLATI, KXLONG )
C****
C Sous-programme servant a specifier des Limites Utilisateur
C en termes de Resolutions horizontale et verticale, valides
C globalement pour tous les Fichiers et Cadres ARPEGE.
C ( Resolution Fichiers - Limites Utilisateur )
C**
C Arguments :       KXNIVV == Nombre maximum de niveaux verticaux;
C (tous d'Entree)   KXTRON == Troncature maximum;
C                   KXLATI == Nombre maximum de latitudes pole a pole;
C                   KXLONG == Nombre maxi de longitudes par parallele.
C*
C S'il y a des cadres deja definis (dynamiquement ou non) avec
C des valeurs correspondantes plus grandes, cela provoque une erreur
```

## fatale

```
      SUBROUTINE FATALE ( KREP, KNUMER, LDERFA )
C****
C Ce sous-programme permet d'activer ou de desactiver l'option
C rendant fatale toute erreur detectee sur un fichier particulier,
C ouvert pour le Logiciel de Fichiers ARPEGE, de meme pour l'option
C correspondante du logiciel LFI.
C Cependant, tant que le niveau global d'erreur fatale *NRFAGA*
C vaut 0 ou 2, l'option propre au fichier est inoperante.
C *NRFAGA* vaut par defaut 1, et est reglable via le s/p "FANERG".
C**
C Arguments : KREP (Sortie) == Code-reponse du sous-programme;
C           KNUMER (Entree) == Numero d'Unite Logique concernee;
C           LDERFA (Entree) == Option d'Erreur Fatale (Vrai=oui).
C

C 2 : Emettre tous les messages possibles, meme s'ils ne correspon-
C     dent pas a un fichier ouvert avec le niveau de Messagerie 2 .
C
```

## fautif

```
      SUBROUTINE FAUTIF ( KREP, KNUMER, CDIDEN )
C****
C Sous-programme permettant de donner un NOM a l'Identificateur
C d'un fichier ARPEGE .
C ( l'Utilisateur Traite son Identificateur de Fichier )
C**
C Arguments : KREP (Sortie) == Code-reponse du sous-programme;
C           KNUMER (Entree) == Numero de l'unite logique;
C           CDIDEN (Entree) == Nom de l'identificateur.
C
C Une messagerie de niveau 1 est emise dans les cas "normaux"
C
```

## faveur

```
      SUBROUTINE FAVEUR ( KREP, KNUMER, KNGRIB, KNBPDG, KNBCSP, KSTRON,
     S                           KPUILA, KDMOPL )
C****
C Ce sous-programme permet d' obtenir, pour un fichier ARPEGE
C ouvert, les options courantes liees au codage GRIB des champs.
C CES OPTIONS NE SONT UTILISEES QUE POUR (RE)ECRIRE DES CHAMPS
C codes en GRIB.
C ( Verification des options Effectives pour l'UtilisateuR )
C**
C Arguments :   KREP == Code-reponse du sous-programme;
C  (tous de   KNUMER == Numero d'Unite Logique concernee;
C  SORTIE)    KNGRIB == Niveau de codage GRIB (0,1,2);
C             KNBPDG == Nombre de bits par valeur point-de-grille;
C             KNBCSP == Nombre de bits par partie reelle/imaginaire
C                         de coefficient spectral;
C             KSTRON == Sous-troncature non compactee;
C             KPUILA == Puissance de laplacien;
C             KDMOPL == Degre de modulation de KPUILA.
C
```

## favori

```
      SUBROUTINE FAVORI ( KNGRIB, KNBPDG, KNBCSP, KSTRON, KPUILA,
     S                           KDMOPL )
C****
C Ce sous-programme permet de connaitre les options implicites
C courantes liees au codage GRIB des champs.
C CES OPTIONS NE SONT UTILISEES QUE POUR (RE)ECRIRE DES CHAMPS
C codes en GRIB, et les valeurs implicites ne servent
C que LORS d'une OUVERTURE de FICHIER.
C ( Verification des Options Reputees Implicites )
C**
C Arguments :   KNGRIB == Niveau de codage GRIB (0,1,2);
C (tous de      KNBPDG == Nombre de bits par valeur point-de-grille;
C  SORTIE)      KNBCSP == Nombre de bits par partie reelle/imaginaire
C                           de coeff. spectral;
C               KSTRON == Sous-troncature non compactee;
C               KPUILA == Puissance de laplacien;
C               KDMOPL == Degre de modulation de KPUILA.
C

C globale.
C Si les valeurs donnees en argument depassent les limites cor-
C repondantes du logiciel, elles sont ecretees, avec un message.
C Une messagerie de niveau 1 est emise dans le cas normal ou
C ci-dessus.
C
```

# <u>5) NOMENCLATURE DES FICHIERS ARPEGE/ALADIN</u>

## 5.1) Fichiers historiques (ou ICMSH)

Par fichier "HISTORIQUE" ou **"ICMSH"** (pour **I**n-**C**ore **M**odel **S**pectral **H**ybrid) on désigne le **<u>principal fichier de données brutes servant d'intermédiaire entre programmes de la chaîne ARPEGE/ALADIN et permettant d'exécuter le modèle de prévision</u>**.

**Par convention**, toutes les lettres des noms de champs sont en **MAJUSCULES.**

**Les sections suivantes décrivent les champs contenus dans ce type de fichier :**

**<u>CHAMPS EN COEFFICIENTS SPECTRAUX</u>** :

La nomenclature proposée pour les champs "dynamiques" nécessaires à un modèle adiabatique humide est la suivante:

- Géopotentiel de surface **(en coefficients spectraux)**: préfixe 'SPECSURF', suffixe 'GEOPOTENTIEL'; niveau arbitraire.

- Pression de surface: préfixe 'SURF', suffixe 'PRESSION'; le niveau est arbitraire.

- Autres champs d'altitude: préfixe 'S', le niveau est compté à partir de 1 "en haut" de l'atmosphère, et les suffixes parlent d'eux-mêmes pour les francophones :

| | |
|---|---|
| 'TEMPERATURE' | Température |
| 'HUMI.SPECIFIQUE' | Humidité specifique |
| 'POT.VITESSE' | Potentiel de vitesse (fichiers ARPEGE) |
| 'FONC.COURANT' | Fonction de courant (fichiers ARPEGE) |
| 'WIND.U.PHYS' | Composante "U" du vent (fichiers ALADIN) |
| 'WIND.V.PHYS' | Composante "V" du vent (fichiers ALADIN). |

Auquels s'ajoutent, dans le cas du modèle Aladin non-hydrostatique, les 2 champs d'altitude suivants :

| | |
|---|---|
| 'VERTIC.DIVER' | Divergence verticale |
| 'PRESS.DEPART' | Ecart de pression par rapport à l'hydrostatique |

N.B. : un suffixe conduisant à un nom d'article trop long est simplement tronqué par le logiciel d'interface, ce qui ne pose aucun problème autre qu'un risque -faible- d'homonymie.

**<u>CHAMPS EN POINTS DE GRILLE</u>** :

En points de grille, les fichiers historiques regroupent des champs climatologiques, des champs pseudo-historiques, des champs historiques ainsi que des champs diagnostiques sous forme de flux instantanés ou cumulés ou instantanés (ces derniers n'étant d'ailleurs pas, en général, des "flux" à proprement parler, mais plutôt des champs).

Champs climatologiques :

| *préfixe* | *suffixe* | *partie tronquée du suffixe* | *Remarques* |
|---|---|---|---|
| SURF | ET.GEOPOTENT | IEL | Ecart-Type ... |
| SURF | IND.TERREMER | | 1. ou 0.<br>(un bit par valeur recommandé!) |
| SURF | PROP.VEGETAT | ION | PROPortion ...<br>(de 0. à 1.) |
| SURF | EMISSIVITE | | |
| SURF | VAR.GEOP.ANI | | Anisotropie du relief sous-maille |
| SURF | VAR.GEOP.DIR | | Angle de la direction du relief avec l'axe des x |
| SURF | IND.VEG.DOMI | | (ISBA) Indice de végétation (de 1. à 5.) |
| SURF | RESI.STO.MIN | | (ISBA) Résistance stomatique minimale |
| SURF | PROP.ARGILE | | (ISBA) |
| SURF | PROP.SABLE | | (ISBA) |
| SURF | EPAIS.SOL | | (ISBA) Epaisseur du réservoir profond |
| SURF | IND.FOLIAIRE | | (ISBA) |
| SURF | GZ0.THERM | | (ISBA) |

Champs pseudo-historiques :

| *préfixe* | *suffixe* | *partie tronquée du suffixe* | *Remarques* |
|---|---|---|---|
| SURF | Z0.FOIS.G | | |
| SURF | ALBEDO | | |
| SURF | RES.EVAPOTRA | | (ISBA) Résistance à l'évapotranspiration |

Champs historiques :

| *préfixe* | *suffixe* | *partie tronquée du suffixe* | *Remarques* |
|---|---|---|---|
| SURF | TEMPERATURE | | |
| PROF | TEMPERATURE | | |
| SURF | RESERV.EAU | | |
| PROF | RESERV.EAU | | |
| SURF | RESERV.GLACE | | |
| PROF | RESERV.GLACE | | |
| SURF | RESERV.NEIGE | | |
| SURF | RESERV.INTER | | (ISBA) Réservoir d'interception |

Flux cumulés :

| *préfixe* | *suffixe* | *partie tronquée du suffixe* | *Remarques* |
|---|---|---|---|
| SURF | PREC.EAU.CON | VECTIVE | Pluies convectives |
| SURF | PREC.EAU.GEC | HELLE | pluies stratiformes |
| SURF | PREC.NEI.CON | VECTIVE | neiges convectives |
| SURF | PREC.NEI.GEC | HELLE | neiges stratiformes |
| SURF | FLU.RAY.THER | MIQUE | |
| SOMM | FLU.RAY.THER | MIQUE | |
| SURF | FLU.RAY.SOLA | IRE | |
| SOMM | FLU.RAY.SOLA | IRE | |
| SURF | FLU.MEVAP.EA | U | |
| SURF | FLU.MSUBL.NE | IGE | |
| SURF | FLU.CHA.SENS | IBLE | |
| SURF | FLU.LAT.MEVA | PORATION | |
| SURF | FLU.LAT.MSUB | LIMATION | |
| SURF | TENS.TURB.ZO | NAL | |
| SURF | TENS.TURB.ME | RIDIEN | |
| SURF | TENS.DMOG.ZO | NAL | Drag Montagneux par Ondes de Gravité |
| SURF | TENS.DMOG.ME | RIDIEN | Drag Montagneux par Ondes de Gravité |

"Flux" instantanés :

| *préfixe* | *suffixe* | *partie tronquée du suffixe* | *Remarques* |
|---|---|---|---|
| SURF | NEBUL.TOTALE | | de 0. à 1. |
| SURF | NEBUL.CONVEC | TIVE | de 0. à 1. |
| SURF | NEBUL.HAUTE | | de 0. à 1. |
| SURF | NEBUL.MOYENN | E | de 0. à 1. |
| SURF | NEBUL.BASSE | | de 0. à 1. |
| CLS | VENT.ZONAL | | |
| CLS | VENT.MERIDIEN | | |
| CLS | TEMPERATURE | | |
| CLS | HUMI.SPECIFIQ | UE | |
| CLS | HUMI.RELATIVE | | |
| CLS | MINI.TEMPERAT | URE | |
| CLS | MAXI.TEMPERAT | URE | |

**REMARQUE** :

Tous les **champs de type flux sont comptés positivement vers le bas**... ce qui explique que devant "Evaporation" et "Sublimation", on ait ajouté un "M" comme "Moins" pour rappeler que le champ en question est majoritairement négatif ou nul.

Les 2 champs CLSMINI.TEMPERATURE et CLSMAXI.TEMPERATURE sont "réinitialisés" après chaque écriture d'un fichier historique; de sorte qu'ils ne représentent les extrema qu'entre deux sorties consécutives de fichiers.

## 5.2) Fichiers de couplage ALADIN

Par "fichiers de couplage" ou fichiers "LBC" (pour Lateral Boundary Conditions) on désigne le fichier de données brutes contenant les conditions aux limites pour exécuter le modèle à aire limitée ALADIN.

Ces fichiers ont le même format et contiennent les mêmes champs que les fichiers historiques (hors flux), quoique les champs de surface physiques contenus y soient inutiles puisque le couplage n'est appliqué que sur les champs d'altitude.

## 5.3) Fichiers de climatologie

Ce type de fichier regroupe des champs de climatologie **comme moyennes mensuelles** nécessaires à la constitution de fichiers historiques. Il est généré par la configuration dite "923" du logiciel ARPEGE/ALADIN.

Les champs **essentiels pour ARPEGE/ALADIN** qui sont contenus dans ce tye de fichier sont les suivants :

| *préfixe* | *suffixe* | *partie tronquée du suffixe* | *Remarques* |
|---|---|---|---|
| SPECSURF | GEOPOTEN | TIEL | En coefficients spectraux |
| SURF | GEOPOTENTIEL | | En points de grille |
| SURF | TEMPERATURE | | |
| PROF | TEMPERATURE | | |
| SURF | PROP.RMAX.EA | U | Proportion du Réservoir de surface MAXimum |
| SURF | PROP.RMAX.EA | U | Proportion du Réservoir profond MAXimum |
| SURF | Z0.FOIS.G | | |
| SURF | ALBEDO | | |
| SURF | RESERV.NEIGE | | |
| SURF | ET.GEOPOTENT | IEL | Ecart-Type ... |
| SURF | IND.TERREMER | | 1. ou 0.<br>(un bit par valeur) |
| SURF | PROP.VEGETAT | ION | PROPortion ...<br>(de 0. à 1.) |
| SURF | EMISSIVITE | | |
| SURF | VAR.GEOP.ANI | | Anisotropie du relief sous-maille |
| SURF | VAR.GEOP.DIR | | Angle de la direction du relief avec l'axe des x |
| SURF | IND.VEG.DOMI | | (ISBA) Indice de végétation (de 1. à 5.) |

| | | | |
|---|---|---|---|
| SURF | RESI.STO.MIN | | (ISBA) Résistance stomatique minimale |
| SURF | PROP.ARGILE | | (ISBA) |
| SURF | PROP.SABLE | | (ISBA) |
| SURF | EPAIS.SOL | | (ISBA) Epaisseur du réservoir profond |
| SURF | IND.FOLIAIRE | | (ISBA) |
| SURF | GZ0.THERM | | (ISBA) |

## 5.4) Fichiers de post-processing ("Fullpos")

Par "fichier de post-processing" ou fichier "Fullpos" on désigne un fichier de données brutes ou élaborées, résultat du post-traitement interne d'ARPEGE ou ALADIN.

Il n'y a pas de règle concernant les champs contenus dans de tels fichiers ; on recommande tout de même, pour l'exploitation, que les champs contenus y soient écrits en points de grille plutôt qu'en spectral.
Quant au format de ces fichiers, il peut être indifféremment :
- une géométrie ARPEGE (grille de Gauss éventuellement basculée/étirée, associée à une troncature spectrale triangulaire),
- une géométrie ALADIN (aire limitée projetée sur un plan, associée à une troncature spectrale elliptique),
- une grille latitude x longitude régulière dans chaque direction (ce format est en fait une géométrie ALADIN possible, mais pas utilisée pour le modèle de prévision).

Par abus de langage, on assimile souvent les fichiers "Fullpos" à ces fichiers de type "latitude x longitude".