

Working Note

WN/CMGC/03/94

Eric MAISONNAVE

Sibylle 1.2: Guide d'utilisation

*Néron ne s'inquiéta pas quand il entendit la prédiction de l'Oracle de Delphes. - Qu'il redoute les
soixante-seize ans*

Constantin Cavafis

Abstract

Sibylle library consists of several scripts written for Ferret visualization software, and fortran/shell external programs. These scripts are mainly developed for Cerfacs Global Change team, in order to ease visualization of data described on irregular grids of OPA and ARPEGE models. Other scripts improve formal graphical functionalities. Others propose mathematical analysis such as density calculation, volumic integral, wavelet analysis, etc. Sibylle library is available on Linux PC, SGI Workstation and Compaq Alphaserwer.

La librairie Sibylle comprend des scripts utilisables par le logiciel de visualisation Ferret, ainsi que des programmes externes Fortran et Shell. Ils ont été écrits dans le but de faciliter la visualisation des données utilisée dans l'équipe Global Change au Cerfacs, en particulier les données au format Netcdf, décrits sur les grilles irrégulières des modèles OPA et ARPEGE. D'autres scripts enrichissent les fonctionnalités de mise en forme des graphiques. D'autres encore permettent d'effectuer dans Ferret des analyses mathématiques, comme des décompositions en ondelettes, des calculs d'intégrales volumiques, des calculs de densité, etc. La librairie Sibylle est disponible sur PC Linux, station SGI et calculateur Compaq Alphaserwer.

Table des matières

1	SIBYLLE VERSION 1.2	1
1.1	<u>Nouveautés:</u>	2
1.2	<u>Configuration:</u>	3
1.3	<u>Palettes:</u>	4
1.4	<u>Scripts:</u>	5
1.4.1	Liste des scripts actuellement disponibles	5
1.4.2	Description fonctionnelle	7
1.4.2.1	anomalie.jnl:	7
1.4.2.2	axe_droit.jnl, axe_gauche.jnl	7
1.4.2.3	densites_orca.jnl:	7
1.4.2.4	ini_arpege.jnl	8
1.4.2.5	ini_orca.jnl	8
1.4.2.6	ini_medias.jnl	8
1.4.2.7	integrale_volume.jnl	9
1.4.2.8	interp_bt42.jnl, inter_medias_globale.jnl, interp_medias_nord_atl.jnl	9
1.4.2.9	grid_orca.jnl	10
1.4.2.10	grid_at42.jnl	10
1.4.2.11	logo_cerfacs.jnl	10
1.4.2.12	make_sland.jnl	11
1.4.2.13	meta2ps.jnl	11
1.4.2.14	plot_at42.jnl	11
1.4.2.15	plot_at42_ortho.jnl et plot_at42_verti.jnl	12
1.4.2.16	plot_at42_polar.jnl	12
1.4.2.17	plot_bt42.jnl	13
1.4.2.18	plot_bt42_ortho.jnl et plot_bt42_verti.jnl	13
1.4.2.19	plot_bt42_polar.jnl	13
1.4.2.20	plot_eof.jnl	14
1.4.2.21	plot_medias_globale.jnl, plot_medias_europe.jnl et plot_medias_france.jnl	14
1.4.2.22	plot_orca.jnl	15
1.4.2.23	plot_orca_ortho.jnl et plot_orca_verti.jnl	15
1.4.2.24	plot_T_S.jnl	16
1.4.2.25	projection_iso.jnl	16
1.4.2.26	sland.jnl	17
1.4.2.27	sym_palette.jnl	17
1.4.2.28	titre.jnl, titre_sur.jnl, titre_sous.jnl	18
1.4.2.29	vector_at42.jnl	18
1.4.2.30	vector_at42_polar.jnl	19
1.4.2.31	vector_bt42.jnl	19
1.4.2.32	vector_bt42_polar.jnl	19
1.4.2.33	vector_medias_france.jnl	20
1.4.2.34	vector_orca.jnl	20
1.4.2.35	vues.jnl	20
1.4.2.36	wavelet.jnl	21
1.4.3	Remarques générales	22
1.4.3.1	Concernant tous les scripts de visualisation	22
1.4.3.2	Concernant les scripts de type *_polar.jnl	22
1.4.3.3	Concernant les scripts de type *_verti.jnl	23
1.4.3.4	Concernant les scripts de type *_ortho.jnl ou *_verti.jnl	23

1.4.3.5	Concernant les scripts de type *_orca.jnl	23
1.4.3.6	Concernant les scripts de visualisation de grilles Arpege:	24

1 SIBYLLE

VERSION 1.2

La librairie Ferret de visualisation des modèles utilisés au CERFACS

Cette librairie regroupe certains scripts développés dans l'équipe Global Change par la très active communauté des utilisateurs Ferret (Carole Cibot, Claire Delon, Marie Drévilion, Andrea Piacentini et Driss Bari). Elle s'appuie sur la librairie de scripts standard Ferret et la librairie Fast (<http://www.ipsl.jussieu.fr/~brocksce/fast/index.html>), développée à l'IPSL par Patrick Brockman. Elle utilise un programme de triangulation de Michel Déqué (CNRM) ainsi que le logiciel de couplage du Cerfacs (OASIS Version 3 en mode interpolateur) développé par Sophie Valcke, Damien Déclat et Arnaud Caubel. Merci également à Olivier Marti (LSCE).

Faites remonter vos remarques et communiquez vos nouveaux scripts (Eric.Maisonnave@cerfacs.fr).
Au Cerfacs, les codes sources se trouvent sur mon HOME:

dans /home/evian/eric/Sibylle.

Ils sont également disponibles sur le site ftp du Cerfacs:

ftp://ftp.cerfacs.fr/pub/globc/exchanges/maisonna/Sibylle_1.2.tar
(ftp://ftp.cerfacs.fr/pub/globc/exchanges/maisonna/Sibylle_1.2.tar)

ainsi que via CVS:

CVSROOT=:pserver:maisonna@elnino.cerfacs.fr:2401/home/evian/ROOTCVS

L'arborescence est organisée comme suit:

Sibylle/			
	Data/		Fichiers netcdf de grilles, poids. Namcouple oasis pour interpolation. Fichier .sib de traits de côtes
	Ext_func/	bin/	Exécutables fortran et scripts shell
		lib/	Fichiers objets issus de la compilation des programmes fortran
		src/	Programmes fortran. Script de compilation
	Go/		Scripts ferret
	Palette/		Palettes ferret
	scripts exferret et exferret_bash		

Note: la librairie est perpétuellement en cours de développement. La page Web (http://www.cerfacs.fr/~maisonna/Sibylle/Sibylle_doc_1_2.html) sera tenue à jour en fonction des nouveaux scripts et de vos remarques.

1.1 Nouveautés:

La librairie Sibylle est désormais disponible sur PC. Elle a été enrichie de nouveaux scripts Ferret, et elle utilise aussi des programmes externes fortran (ou scripts shell).

Une nouvelle grille est visualisable (la grille stretchée d'Arpege). Toutes les échéances temporelles d'un fichier en grille réduite Arpege sont désormais plus facilement manipulables grâce à leur interpolation par le logiciel OASIS Version 3 et leur sauvegarde dans un fichier Netcdf. D'autres modifications enrichissent la mise en forme des données visualisées.

1.2 Configuration:

La version Ferret5.41 (ou Ferret5.51 sur linux) a été installée sur toutes les stations de travail du Cerfacs, les PC Linux, ainsi que sur l'Alphaserver Compaq "imhotep". Elle est indispensable pour pouvoir visualiser des champs de vecteurs sur les grilles non régulières ORCA et Arpege BT42.

Pour lancer Ferret avec les variables d'environnement permettant de se servir de Sibylle, créez un script de lancement à partir des scripts fournis (Sibylle/exferret ou Sibylle/exferret_bash si votre shell courant \$SHELL est bash) que vous déplacerez, par exemple, dans votre répertoire ~/bin (répertoire compris dans votre PATH).

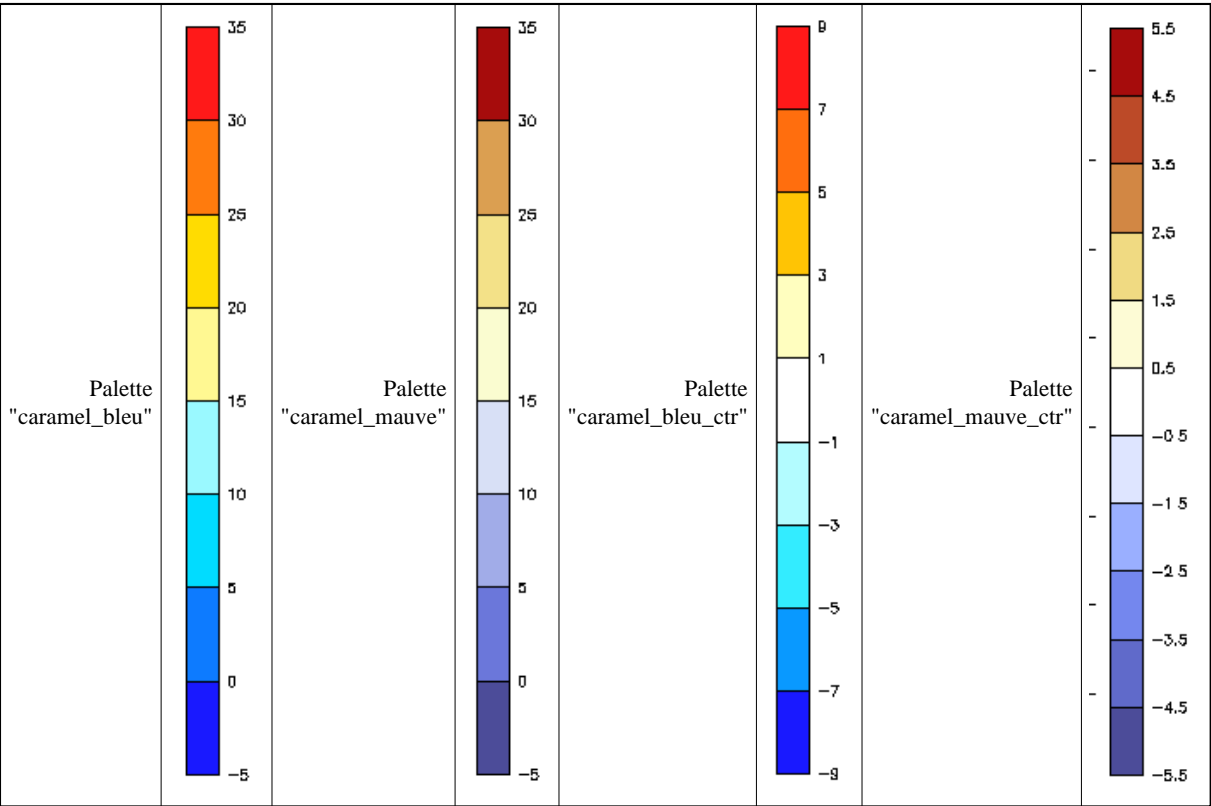
Sur les stations et tx, dans ce script exferret, modifier la chaîne de caractères **_your_Ferret_path_** par **/usr/local/ferret541/paths/ferret_paths** et la chaîne de caractères **_your_Sibylle_path_** par **/home/evian/eric/Sibylle**.

Sur les PC, modifier la chaîne de caractères **_your_Ferret_path_** par **/usr/local/ferret551/paths/ferret_paths** et la chaîne de caractères **_your_Sibylle_path_** par **/home/evian/eric/Sibylle**.

Sur la machine de calcul imhotep, modifier la chaîne de caractères **_your_Ferret_path_** par **/usr/local/ferret541/paths/ferret_paths** et la chaîne de caractères **_your_Sibylle_path_** par **/home/globc_varia/varia/maisonna/Sibylle**.

1.3 Palettes:

Quatre palettes créées au Cerfacs sont disponibles en plus des palettes standard de Ferret. Pour des tracés "centrés", utiliser les deux dernières palettes



Utilisation: "PALETTE caramel_bleu" ou "PALETTE caramel_mauve"

1.4 Scripts:

1.4.1 *Liste des scripts actuellement disponibles*

Initialisation des paramètres de grille:

ORCA	ini_orca.jnl
Arpege	ini_arpege.jnl
Medias (Arpege étiré)	ini_medias.jnl

Visualisation grilles 2D:

	Grille globale	Projection orthographique	Projection verticale	Projection polaire
ORCA	plot_orca.jnl	plot_orca_ortho.jnl	plot_orca_verti.jnl	
Arpege AT_42	plot_at42.jnl	plot_at42_ortho.jnl	plot_at42_verti.jnl	plot_at42_polar.jnl
Arpege BT_42	plot_bt42.jnl	plot_bt42_ortho.jnl	plot_bt42_verti.jnl	plot_bt42_polar.jnl
Medias	plot_medias_globale.jnl plot_medias_europe.jnl plot_medias_france.jnl			

Visualisation vecteur 2D:

	Grille globale	Projection polaire
ORCA	vector_orca.jnl	
Arpege AT_42	vector_at42.jnl	vector_at42_polar.jnl
Arpege BT_42	vector_bt42.jnl	vector_bt42_polar.jnl
Medias	vector_medias_france.jnl	

Interpolation grilles 2D:

	Grille globale
Arpege BT_42	interp_bt42.jnl
Medias	interp_medias_globale.jnl interp_medias_nord_atl.jnl

Autres scripts de visualisation:

Diagramme T_S	plot_T_S.jnl
Tracé d'iso-surfaces	projection_iso.jnl
Tracé d'EOF (pour sorties Statpack)	plot_eof.jnl
Tracé d'une analyse en ondelette	wavelet.jnl

Scripts de mise en forme graphique:

Titres	titre.jnl , titre_sur.jnl, titre_sous.jnl
Redéfinition des sous-fenêtres (view)	vues.jnl
Logo du Cerfacs	logo_cerfacs.jnl
Superposition du maillage: grille ORCA/grille Arpege	grid_orca.jnl grid_at42.jnl
Création de fonds de cartes	make_sland.jnl
Superposition de fonds de cartes	sland.jnl
Ajustement des min/max à une palette de couleurs symétriques par rapport à un zéro	sym_palette.jnl
Conversion fichier metafile.plt en postscript	meta2ps.jnl
Axes d'ordonnées droit et gauche	axe_droit.jnl, axe_gauche.jnl

Scripts de calculs:

Anomalies	anomalie.jnl
Densités	densites_orca.jnl
Intégrale volumique	integrale_volume.jnl

1.4.2 Description fonctionnelle

1.4.2.1 anomalie.jnl:

Fonction: Calculer la moyenne d'un champ d'un fichier donné ainsi que l'anomalie à cette climatologie. Retourner ces deux valeurs dans deux variables ferret

ENTREE: Argument1 : Nom du champ 2D ou 3D

SORTIE: Variables *anomalie_\$Champ* et *climatologie_\$Champ*

Exemple: GO anomalie SOSSTSST

Précautions d'usage: ne rien spécifier entre crochets [] dans l'argument 1 "Champ" (calcul pour tout i,j,k et l). Pour réduire les calculs à des domaines, utiliser la fonction "set region".

1.4.2.2 axe_droit.jnl, axe_gauche.jnl

Fonction: Visualiser deux courbes d'ordonnées différentes sur un même graphique

ENTREE: Argument1 : Nom du champ 1D ...

Exemple: SET REGION/L=0:200
GO axe_gauche IICETHIC
GO axe_droit ISNOWTHI/1000

Remarque: .les fonctions axe_droit et axe_gauche ont les mêmes arguments que les scripts ferret right_axis_plot et left_axis_plot concernant les labels, couleurs, etc ...

1.4.2.3 densites_orca.jnl:

Fonction: A partir des temperature et salinités en sortie du modèle OPA, calculer la densité in-situ et potentielle selon deux méthodes. Retourner ces trois valeurs dans des variables ferret

- 1ère méthode: à partir de l'équation d'état utilisée dans ORCA (Jackett and McDougall (1994)) calcul de la densité in-situ: $\rho(t,s,p)$ et potentielle $\rho(t,s,0)$ (variables *densite_in_situ* et *masse_vol_potentielle*).
- 2ème méthode (plus rapide): à partir de l'équation d'état donnée en standard dans ferret (1980 UNESCO International Equation of State, Millero and Poisson). (variable *densite_ferret*)

ENTREE: Pas d'argument. Un fichier ORCA contenant salinités et températures doit cependant avoir été chargé au préalable.

SORTIE: Variables *densite_in_situ*, *masse_vol_potentielle* et *densite_ferret*

Exemple: GO densites_orca

1.4.2.4 ini_arpege.jnl

Fonction: Charger des fichiers de grille et de masque de la grille ARPEGE pour visualisation
Cette fonction charge ("use" ferret) les fichiers de grilles (grids.des) et de masques (masks.des) des grilles des modèles Orca et Arpege et pré-définit de nouvelles variables (latitudes, longitudes, masques) utilisées dans les scripts de classe "plot_" et "vector_".

Ces fichiers sont localisés dans le répertoire \$FER_DATA.

La plupart des fonctions de visualisation de notre bibliothèque charge et décharge ces données automatiquement. L'utilisateur n'a donc pas à lancer cette commande dans le cas d'une utilisation standard des outils SIBYLLE.

ENTREE: pas d'argument

SORTIE: Variable **new** sur la
grille **AT42_GRID**

Exemple: GO ini_arpege

Remarque: les grilles sont chargées, mais le fichier de données par défaut n'est pas modifié par le lancement de la commande. Pour utiliser les champs des fichiers grids.des et masks.des, utiliser l'option [**d=masks**] ou [**d=grids**] à chaque appel d'un champ de masque ou de grille.

Le déchargement des fichiers de données peut se faire en lançant les commandes: "cancel data masks" et "cancel data grids"

1.4.2.5 ini_orca.jnl

Fonction: Charger des fichiers de grille et de masque de la grille ORCA pour visualisation
voir "ini_arpege.jnl"

ENTREE: pas d'argument

SORTIE: Variables **lon2, lat2**: tableaux 2D (182x149) des longitudes et latitudes de chaque point de la grille ORCA, modifiées pour visualisation.
Variable masquée **TORC_MSK[d=masks]**

Exemple: GO ini_orca

Remarque: voir ini_arpege.jnl.

La variable **TORC_MSK** est un masque terre qui peut être appliqué à un champ ORCA ne contenant pas de masque (fichier de forçage, résultat d'interpolation, grille Levitus ...). Pour cela, on visualisera le champ "CHAMP+TORC_MSK[d=masks]".

1.4.2.6 ini_medias.jnl

Fonction: Charger des fichiers de grille et de masque de la grille MEDIAS pour interpolation et des grilles régulières 1 degrés, 0.5 degrés et 0.1 degrés pour visualisation
voir "ini_arpege.jnl"

ENTREE: pas d'argument

SORTIE: Variables **lon_xxxx,lat_xxxx[d=reg_grids]**: tableaux 2D des longitudes et latitudes de chaque point des grilles régulières (avec xxxx=1010 pour la grille 1 degrés, 0505 pour la grille 0.5 degrés et 0101 pour la grille 0.1 degrés)
 Variables **MEGL_LON, MEGL_LAT, MEFR_LON, MEFR_LAT[d=grids]**: tableaux 2D des longitudes et latitudes de chaque point des grilles MEDIAS (complète (sur 21692 points) et réduite sur la France (sur 360 points))
 Variable masquée **MEGL_MSK[d=masks]**. La grille MEDIAS sur 360 points ne compte que des points non masqués
 Variable **new_xxxx** sur la grille régulière xxxx

Exemple: GO ini_medias

Remarque: voir ini_arpege.jnl.

La variable **MEGL_MSK** est un masque terre qui peut être appliqué à un champ MEDIAS global ne contenant pas de masque. Pour cela, on visualisera le champ "CHAMP+MEGL_MSK[d=masks]".

1.4.2.7 integrale_volume.jnl

Fonction: calcul de l'intégrale d'un champ volumique entre deux isosurfaces

ENTREE: Argument 1: Nom du champ à intégrer (3D volumique + dimension temporelle éventuelle)
 Argument 2: Nom du champ sur lequel se trouvent les isosurfaces (3D volumique + dimension temporelle éventuelle)
 Argument 3: Valeur de l'isosurface inférieure
 Argument 4: Valeur de l'isosurface supérieure

SORTIE: fichier int_vol.nc, variable: *vol_field*

Exemple: GO integrale_volume VOTEMPER VORHOPOT 1024 1025.5

Dans cet exemple, on calcule une champ 2D + dimension temporelle, nommé "vol_field", contenu dans un fichier int_vol.nc. Ce champ représente l'évolution temporelle de l'intégrale de la température dans un volume compris entre l'isopycne 1024 et l'isopycne 1025.5.

Remarque: les calculs étant un peu longs, une réduction du domaine de calcul est recommandée. On spécifiera les limites dans la variable "champ à intégrer", la limitation se faisant alors automatiquement sur le champ des isosurfaces.

Exemple: GO integrale_volume VOTEMPER[i=50:60,j=30:40] VORHOPOT 1024 1025.5

- DROITS D'ECRITURE

1.4.2.8 interp_bt42.jnl, inter_medias_globale.jnl, interp_medias_nord_atl.jnl

Fonction: interpolation de champs 2D + dimension temporelle sur une grille régulière

Les champs des grilles réduites et stretchées Arpege peuvent être visualisées par l'intermédiaire des scripts plot_bt42 et plot_medias. Mais on ne peut visualiser qu'une échéance temporelle à la fois et la manipulation de champs provenant de plusieurs échéances temporelles est parfois problématique.

Les fonctions "interp" offrent donc la possibilité d'interpoler l'ensemble d'un fichier netcdf

2D+dimension temporelle dans un fichier nommé a_interp.nc ou a_trian.nc, sur une grille régulière:
grille régulière AT42 Arpege pour interp_bt42
grille régulière globale 1degré x 1degré pour interp_medias_globale
grille régulière réduite à l'Atlantique Nord 0.5 degrés x 0.5 degrés pour interp_medias_nord_atl

ENTREE: Argument 1: Nom du champ à interpoler (2D surfacique + dimension temporelle éventuelle)

SORTIE: fichier a_interp.nc ou a_trian.nc, variable: *a_interp* ou *a_trian*

Exemple: GO interp_medias_nord_atl PLMEASLP
SHADE a_trian[l=@ave]

- DROITS D'ECRITURE

1.4.2.9 grid_orca.jnl

Fonction: Superposer à un champ ORCA le maillage de la grille

ENTREE: Argument 1: Step (dessiner le maillage tous les "step" points de grille) (par défaut: 5)

Exemple: GO grid_orca 1

Remarque: Grille T seulement. Ne fonctionne qu'en projection non polaire.

1.4.2.10 grid_at42.jnl

Fonction: Superposer à un champ "Arpege régulier" le maillage de la grille.

ENTREE: Argument 1: Step (dessiner le maillage tous les "step" points de grille) (par défaut: 5)

Exemple: GO grid_at42 1

Remarque: Ne fonctionne qu'en projection non polaire.

1.4.2.11 logo_cerfacs.jnl

Fonction: Faire apparaître le logo du cerfacs (coin supérieur droit) sur un graphe

ENTREE: pas d'argument. Utilise le fichier logo_cerfacs.nc et la palette cerfacs.spk

SORTIE: Variable définissant une zone de fenêtre (view) : LogoR

Exemple: GO logo_cerfacs

Précautions d'usage: Lancer cette commande avant toute commande de définition de vue ("set view").

Remarque: Pour créer un nouveau logo, utiliser la fonction /home/dasp/maisonna/ferret/predicate/New_scripts/mak_logo.jnl

1.4.2.12 make_sland.jnl

Fonction: Créer des fonds de carte à superposer dans les visualisations 2D

Sur le site Web <http://crusty.er.usgs.gov/coast/getcoast.html>, on peut sélectionner une région du globe et télécharger le fichier ASCII de traits de côtes correspondant. Le programme make_sland permet de convertir ce fichier au format Sibylle pour qu'il puisse être visualisé via la fonction sland.jnl.

ENTREE: Argument 1: nom du fichier ASCII de traits de côtes

SORTIE: Fichier .sib à copier

Exemple: GO make_sland egee.dat

Remarque: Le fichier .sib obtenu doit être copié dans le répertoire \$FER_SIBYLLE/Data pour être toujours trouvé par le script sland.jnl.

1.4.2.13 meta2ps.jnl

Fonction: Transformer le fichier metafile.plt en un fichier postscript

ENTREE: Argument 1: nom du fichier postscript
Argument 2: format Portrait (P) ou
Paysage (L)

SORTIE: Fichier .sib à copier

Exemple: GO meta2ps egee.ps L

Remarque: Le fichier metafile s'obtient toujours en entourant les lignes de commandes Ferret de tracé par les commandes "set mode metafile" et "cancel mode metafile".

- DROITS D'ECRITURE

1.4.2.14 plot_at42.jnl

Fonction: Visualiser un champ at42 (2D) en projection globale

ENTREE: Argument 1: Nom du champ
Argument 2: Commande du plot avec option (ex: shade/line/levels) (par défaut: shade)
Argument 3: Masque: TERRE ou MER ou NONMASQUE (par défaut: NONMASQUE)

Exemple: GO plot_at42 PLZOW200[L=1] fill/line TERRE

Remarque:

- DIMENSION TEMPORELLE

1.4.2.15 plot_at42_ortho.jnl et plot_at42_verti.jnl

Fonction: Visualiser un champ at42 (2D) en projection orthographique ou verticale

ENTREE: Argument 1: Nom du champ

Argument 2: Parallèle central (par défaut : 90 -pole Nord-)

Argument 3: Méridien Central (par défaut : 0 -Greenwich-)

Argument 4: Commande du plot avec option (ex: shade/line/levels) (par défaut: shade)

Argument 5: Masque: TERRE ou MER ou NONMASQUE (par défaut: NONMASQUE)

Exemple: GO plot_at42_ortho PLZOW200[L=1] -40 80 fill/line TERRE

Remarques:

- DISTANCE VERTICALE
- CONTINENTS EN PROJECTION ORTHO OU VERTICALE
- DIMENSION TEMPORELLE

1.4.2.16 plot_at42_polar.jnl

Fonction: Visualiser un champ at42 (2D) en projection polaire équidistante

ENTREE: Argument 1: Nom du champ

Argument 2: Pôle Nord (N) ou Sud (S) (par défaut : N-pôle Nord-)

Argument 3: Méridien Central (entr e-180 et 180) (par défaut : 0 -Greenwich-)

Argument 4: Commande du plot avec option (ex: shade/line/levels) (par défaut: shade)

Argument 5: Masque: TERRE ou MER ou NONMASQUE (par défaut: NONMASQUE)

Exemple: GO plot_at42_polar PLZOW200[L=1] N 80 fill/line TERRE

Remarques:

- DROITS D'ECRITURE
- CONTINENTS, CONTOUR DES GRILLES EN PROJECTION POLAIRE
- DIMENSION TEMPORELLE

1.4.2.17 plot_bt42.jnl

Fonction: Visualiser un champ bt42 (2D) en projection globale

Ce script procède à une projection du vecteur 1D de 6232 points de la grille BT42 sur les points de la grille AT42 (fonction ferret scat2gridgauss_XY).

Sur les bandes de latitude élevée, certains points de la grille AT42 ne sont pas remplis par des valeurs de la grille BT42. On effectue donc une deuxième interpolation, linéaire et suivant X (commande let trace = samplebt[i=@fln]).

L'interpolation scat2gridgauss_XY ne tient pas compte de l'option [l=...] fixant la coordonnée temporelle. On décale donc les données d'entrée à l'aide de l'option [L=@shf:(\$secheance)] ou \$secheance est fixée par l'utilisateur en troisième argument de la fonction.

ENTREE: Argument 1: Nom du champ
Argument 2: Commande du plot avec option (ex: shade/line/levels) (par défaut: shade)
Argument 3: Coordonnée temporelle (par défaut L=1)
Argument 4: Masque: TERRE ou MER ou NONMASQUE (par défaut: NONMASQUE)

Exemple: GO plot_bt42 COWATFLU fill/line 5 TERRE

Remarque:

- DIMENSION TEMPORELLE

1.4.2.18 plot_bt42_ortho.jnl et plot_bt42_verti.jnl

Fonction: Visualiser un champ bt42 (2D) en projection orthographique ou verticale

ENTREE: Argument 1: Nom du champ
Argument 2: Parallèle central (par défaut : 90 -pôle Nord-)
Argument 3: Méridien Central (par défaut : 0 -Greenwich-)
Argument 4: Commande du plot avec option (ex: shade/line/levels) (par défaut: shade)
Argument 5: Coordonnée temporelle (par défaut L=1)
Argument 6: Masque: TERRE ou MER ou NONMASQUE (par défaut: NONMASQUE)

Exemple: GO plot_bt42_ortho COWATFLU 90 70 shade/line 5 MER

Remarques:

- DISTANCE VERTICALE
- CONTINENTS EN PROJECTION ORTHO OU VERTICALE
- DIMENSION TEMPORELLE

1.4.2.19 plot_bt42_polar.jnl

Fonction: Visualiser un champ bt42 (2D) en projection polaire équidistante

Argument 1: Nom du champ
Argument 2: Pôle Nord (N) ou Sud (S) (par défaut : N-pôle Nord-)
ENTREE: Argument 3: Méridien Central (entre -180 et 180) (par défaut : 0 -Greenwich-)
Argument 4: Commande du plot avec option (ex: shade/line/levels) (par défaut: shade)
Argument 5: Coordonnée temporelle (par défaut L=1)
Argument 6: Masque: TERRE ou MER ou NONMASQUE (par défaut: NONMASQUE)

Exemple: GO plot_bt42_polar SISUTESU N -90 shade/line 5 TERRE

Remarques:

- DROITS D'ECRITURE
- CONTINENTS, CONTOUR DES GRILLES EN PROJECTION POLAIRE
- DIMENSION TEMPORELLE

1.4.2.20 plot_eof.jnl

Fonction: Visualiser deux modes EOF (pattern + composantes principales) d'un fichier de sortie de Statpack.

Le pourcentage de variance expliquée par chaque mode est également affiché (en titre du graphe des composantes principales)

ENTREE: Argument 1: Nom du champ
Argument 2: Numéro du premier mode à visualiser (defaut:1)
Argument 3: Numéro du second mode à visualiser (defaut:1)
Argument 4: Commande du plot avec option (ex: shade/line/levels)
(par défaut: plot_orca)

Exemple: GO plot_eof SOSSTSST 1 2 fill

1.4.2.21 plot_medias_globale.jnl, plot_medias_europe.jnl et plot_medias_france.jnl

Fonction: Visualiser un champ medias (2D) en projection globale

Le script plot_medias_france permet de visualiser les 360 points de la grille Medias France en les interpolant sur une grille régulière 0.1 degrés.

Les deux autres scripts permettent de visualiser les champs de la grille complète Medias (Arpege étiré) en l'interpolant soit sur une grille de 0.5 degrés de côté (pour une visualisation sur l'Europe) ou sur une grille de 1 degré de côté (pour une vue plus globale).

Ces scripts utilisent la fonction Ferret scat2gridgauss_XY (voir plot_bt42.jnl)

Sur la visualisation globale, deux interpolations linéaires sont faites suivant les axes I et J.

ENTREE: Argument 1: Nom du champ
 Argument 2: Commande du plot avec option (ex: shade/line/levels)
 (par défaut: shade)
 Argument 3: Coordonnée temporelle (par défaut L=1)
 Argument 4 (pour plot_medias_globale et plot_medias_europe seulement) : Masque: TERRE ou MER ou NONMASQUE
 (par défaut: NONMASQUE)

Exemple1: GO plot_medias_globale SUTOPRSU shade 1
 Exemple2: GO plot_medias_europe SUTOPRSU shade 1 TERRE
 Exemple3: GO plot_medias_france SITEMPCL fill 1

1.4.2.22 plot_orca.jnl

Fonction: Visualiser un champ ORCA (2D) en projection globale

ENTREE: Argument 1: Nom du champ
 Argument 2: Commande du plot avec option (ex: shade/line/levels) (par défaut: shade)

Exemple: GO plot_orca SOSSTSST[L=1] fill/line

Remarques:

- MASQUES
- ZOOM

1.4.2.23 plot_orca_ortho.jnl et plot_orca_verti.jnl

Fonction: Visualiser un champ ORCA (2D) en projection orthographique ou verticale

ENTREE: Argument 1: Nom du champ
 Argument 2: Parallèle central (par défaut : 90 -pôle Nord-)
 Argument 3: Méridien Central (par défaut : 0 -Greenwich-)
 Argument 4: Commande du plot avec option (ex: shade/line/levels) (par défaut: shade)

Exemple: GO plot_orca_ortho SOSSTSST[L=1] 90 70 fill/line

Remarques:

- DISTANCE VERTICALE
- CONTINENTS EN PROJECTION ORTHO OU VERTICALE
- MASQUES
- ZOOM

1.4.2.24 plot_T_S.jnl

Fonction: Tracé d'un diagramme T/S

ENTREE: Argument 1: Nom du champ de température
Argument 2: Nom du champ de salinité
Argument 3: Argument de la commande plot (par défaut: nolab)

Exemple: GO plot_T_S

```
SOSSTSST[i=114:120@ave,j=114:120@ave]  
SOSALINE[i=114:120@ave,j=114:120@ave]  
pen=3
```

Remarque:

- La croix matérialise le premier plot T/S (celui de la première coordonnée temporelle choisie).
- Passer deux champs 1D dans les deux premiers arguments.
- On peut ne tracer qu'une échéance temporelle sur 12, par exemple. On précise alors *nom_champ*[l=1:120:12]. Si l'on souhaite commencer à une coordonnée temporelle différente de 1, on écrira [l=N:120:12].

1.4.2.25 projection_iso.jnl

Fonction: Projection d'un champ ORCA 3D sur l'iso-surface d'un autre

ENTREE: Argument 1: Nom du champ à projeter
Argument 2: Nom du champ sur lequel projeter
Argument 3: Valeur définissant l'iso-surface sur laquelle projeter le premier champ
Argument 4: Argument de la commande plot (par défaut: shade)

Exemple: GO projection_iso VOTEMPER[L=1] VOSALINE[L=1] 35 fill

Remarques:

- Le champ est visualisé sur la grille ORCA
- En combinant les fonctions *densites_orca* et *projection_iso*, on peut, par exemple, visualiser les températures d'une isopycne:

```
go densites_orca puis  
go projection_iso VOTEMPER[L=1] masse_vol_potentielle[l=1] 1025
```

1.4.2.26 sland.jnl

Fonction: Faire apparaître le contour des continents sur une visualisation d'un champ 2D
Utilisé avec l'argument "orca", la fonction sland superpose un fond de carte global à la grille ORCA.
La fonction standard "land" superpose le contour des continents sur une plage de longitude allant de -180 à 750 degrés.
La grille ORCA commençant à -275 degrés, le script "land.jnl" d'origine a été transformé: les contours des continents ont été traduits de -360 degrés.
Cette fonction permet également de visualiser les fonds de cartes créés avec la fonction make_sland.jnl. On donne alors comme argument à la fonction le nom du fichier .sib correspondant (sans l'extension .sib).
Avec l'argument "list", la fonction renvoie l'ensemble des fonds de cartes disponibles dans le répertoire \$FER_SIBYLLE/Data.

Argument 1:

- "orca" pour superposer un fond de carte global à un champ ORCA
- "nom_fond_de_carte_Sibylle" pour superposer un fond de carte créé avec make_sland.jnl

ENTREE: • "list" pour lister les différents fonds de carte Sibylle disponibles

Argument 2: "overlay" (superposition) ou "basemap" (carte de fond)

Argument 3: (avec argument 1 "orca" seulement) Couleur des contours des pays

Argument 4: (avec argument 1 "orca" seulement) Couleur des contours des états d'un certain pays ... mais lequel ? A toi de trouver, petit ami !

Exemple: GO sland iroise basemap

1.4.2.27 sym_palette.jnl

Fonction: Ajustement symétrique d'une palette par rapport à zéro ou à un niveau donné
Pour séparer, de façon immédiatement lisible, les valeurs d'un champ 2D en deux parties distinctes (par exemple la partie négative de la partie positive), on utilise une palette de couleurs symétriques. Par exemple, pour la palette Sibylle "caramel_bleu", des teintes rouges colorent la partie positive, des bleues la partie négative.
La fonction sym_palette permet de centrer la palette autour d'un zéro (0 ou valeur fixée par l'utilisateur).

ENTREE: Argument 1: Champ
Argument 2: Valeur du niveau "zéro"

Exemple: GO sym_palette SOSSHEIG[l=1]
niveaux de la palette)
puis SET view full (définition de la vue du tracé)
puis SHADE/lev SOSSHEIG[l=1]
niveaux dans la commande de tracé)

Remarque:

Cette fonction a besoin de redéfinir temporairement une "vue" ferret. C'est pourquoi il est impératif de redéfinir la vue dans laquelle le graphe à palette symétrique va être dessiné. On lance ensuite la commande de tracé en ajoutant l'option /lev (pour reprendre les niveaux précédemment définis par la fonction sym_palette).

1.4.2.28 titre.jnl, titre_sur.jnl, titre_sous.jnl

Fonction: Ajouter d'un titre à un graphique

Ce graphique doit avoir été composé avec une commande comprenant l'option /nolabel, ce qui est le cas de toutes les commandes de visualisation SIBYLLE. Ce script utilise la commande ferret "label" en recalculant la position en X et en fixant la position Y (-.7), le mode de justification (0, centrée), l'angle (0, horizontal) et la taille (.2 pouces).

"titre_sous.jnl" fixe, lui, la position en Y à un cinquième de la taille totale du graphique en dessous de celui-ci (même option que titre.jnl).

"titre_sur.jnl" permet quand à lui de placer au dessus du graphique un titre et, éventuellement, un sous titre.

ENTREE: Argument 1: Chaîne de caractère du titre
 Argument 2: Taille de la chaîne (défaut: 0.2)
 Argument 3 (pour titre_sur seulement): Chaîne de caractère du sous-titre
 Argument 4: Taille de la chaîne sous-titre (défaut: 0.2)

Exemple: GO titre "Précipitations totales"

Remarque:

- Ne pas oublier les guillemets si votre titre comprend des espaces. Pour afficher une apostrophe, en écrire 2: "'".

1.4.2.29 vector_at42.jnl

Fonction: Visualiser un champ de vecteurs at42 (2D) en projection globale

ENTREE: Argument 1: Nom du champ zonal
 Argument 2: Nom du champ méridien
 Argument 3: Argument de la commande vector (ex: aspect) (par défaut: nolabel)
 Argument 4: Masque: TERRE ou MER ou NONMASQUE (par défaut: NONMASQUE)

Exemple: GO vector_at42 PLZOW200 PLMEW200 overlay/l=1 TERRE

Remarque:

- DIMENSION TEMPORELLE

1.4.2.30 vector_at42_polar.jnl

Fonction: Visualiser un champ de vecteurs at42 (2D) en projection polaire équidistante

ENTREE: Argument 1: Nom du champ zonal
Argument 2: Nom du champ méridien
Argument 3: Pôle Nord (N) ou Sud (S) (par défaut : N-pôle Nord-)
Argument 4: Méridien Central (entre -180 et 180) (par défaut : 0 -Greenwich-)
Argument 5: Argument de la commande vector (ex: aspect) (par défaut: nolabel)
Argument 6: Masque: TERRE ou MER ou NONMASQUE (par défaut: NONMASQUE)

Exemple: GO vector_at42_polar PLZOW200 PLMEW200 S 70 aspect/L=1 TERRE

Remarque:

- Les options "len=20/xskip=10/yskip=10/pen=1" ont été optimisées et fixées en dur dans le script.
- DROITS D'ECRITURE
- CONTINENTS, CONTOUR DES GRILLES EN PROJECTION POLAIRE
- DIMENSION TEMPORELLE

1.4.2.31 vector_bt42.jnl

Fonction: Visualiser un champ de vecteurs bt42 (2D) en projection globale

ENTREE: Argument 1: Nom du champ zonal
Argument 2: Nom du champ méridien
Argument 3: Argument de la commande vector (ex: aspect) (par défaut: nolabel)
Argument 4: Coordonnée temporelle (par défaut L=1)
Argument 5: Masque: TERRE ou MER ou NONMASQUE (par défaut: NONMASQUE)

Exemple: GO vector_bt42 SIZOWICL SIMEWICL aspect 120 TERRE

Remarque:

- DIMENSION TEMPORELLE

1.4.2.32 vector_bt42_polar.jnl

Fonction: Visualiser un champ de vecteurs at42 (2D) en projection polaire équidistante

ENTREE: Argument 1: Nom du champ zonal
Argument 2: Nom du champ méridien
Argument 3: Pôle Nord (N) ou Sud (S) (par défaut : N-pôle Nord-)
Argument 4: Méridien Central (entre -180 et 180) (par défaut : 0 -Greenwich-)
Argument 5: Argument de la commande vector (ex: aspect) (par défaut: nolabel)
Argument 6: Coordonnée temporelle (par défaut L=1)
Argument 7: Masque: TERRE ou MER ou NONMASQUE (par défaut: NONMASQUE)

Exemple: GO vector_bt42_polar SIZOWICL SIMEWICL S 90 noaxis 30 TERRE

Remarques:

- DROITS D'ECRITURE
- CONTINENTS, CONTOUR DES GRILLES EN PROJECTION POLAIRE
- DIMENSION TEMPORELLE
- voir vector_at42_polar.jnl

1.4.2.33 vector_medias_france.jnl

Fonction: Visualiser un champ de vecteurs Medias (2D) en projection centrée sur la France

ENTREE: Argument 1: Nom du champ zonal
Argument 2: Nom du champ méridien
Argument 3: Argument de la commande vector (ex: aspect) (par défaut: nolabel)
Argument 4: Coordonnée temporelle (par défaut L=1)
Argument 5: Masque: TERRE ou MER ou NONMASQUE (par défaut: NONMASQUE)

Exemple: GO vector_medias_france SIZOWICL SIMEWICL aspect 120

1.4.2.34 vector_orca.jnl

Fonction: Visualiser un champ de vecteurs ORCA (2D) en projection globale

ENTREE: Argument 1: Nom du champ zonal
Argument 2: Nom du champ méridien
Argument 3: Argument de la commande vector (ex: aspect) (par défaut: nolabel)

Exemple: GO vector_orca SOZOTAUX[d=1] SOMETAUUY[d=2] aspect/L=1

Remarques:

- MASQUES
- ZOOM

1.4.2.35 vues.jnl

Fonction: Définir de nouvelles zones de fenêtres (view)

Ces nouvelles vues permettent de mieux utiliser l'espace de la fenêtre avec des graphiques sans labels. L'appel préalable à ces fonctions de définitions de vues est donc fortement conseillé avec l'utilisation des fonctions de visualisation de la librairie SIBYLLE.

ENTREE: Argument 1: Nombre de vues par colonne (entre 2 et 4)
 Argument 2: Nombre de vues par ligne (entre 1 et 2)
 Argument 3: Orientation (portrait (P) ou paysage (L))

SORTIE: Variables définissant une zone de fenêtre (view) : v[1-8]

Exemple: GO vues 3 2 P

1.4.2.36 wavelet.jnl

Fonction: Calculer et afficher une analyse en ondelettes sur une série temporelle 1D.

La description complète de cette fonction se trouve sur le site:

<http://paos.colorado.edu/research/wavelets/> (<http://paos.colorado.edu/research/wavelets/>)

ENTREE: Argument 1: Nom de la variable 1D temporelle
 Argument 2: Chaîne de caractères du titre (défaut: "Variable (Argument 1)")
 Argument 3: Comparaison du spectre avec un bruit rouge ou un bruit blanc (défaut: rouge)
 Argument 4: Pas de l'échantillonnage (période de la série temporelle) (défaut: 1)
 Argument 5: Echelle minimum de l'ondelette (défaut: pas * 2)
 Argument 6: Ondelette mère: 0= Morlet (défaut), 1= Paul, 2= DOG (Derivative Of Gaussian)
 Argument 7: Paramètre de l'ondelette mère:

 Pour Morlet, k0 (nombre d'onde), (défaut: 6)
 For Paul, m (ordre), (défaut: 4)
 For DOG , m (dérivée m-ième), (défaut: 2)

 Argument 8: Deuxième Champ 1D (défaut: aucun)
 Argument 9: Titre du deuxième champ (défaut: aucun)
 Argument 10: Comparaison du spectre avec un bruit rouge ou un bruit blanc (défaut: rouge)

Exemple: GO wavelet nino3_index

- DROITS D'ECRITURE

1.4.3 Remarques générales

1.4.3.1 Concernant tous les scripts de visualisation

ARGUMENTS:

Pour donner à un de ces scripts un argument qui ne serait pas le premier, l'utilisateur doit spécifier TOUS les arguments précédents ou les remplacer par un espace entouré de deux guillemets: " ".

USE:

Ces scripts s'utilisent comme des commandes shade ou vector. Les fichiers de données doivent être chargés avant lancement des commandes.

Les numéros de fichiers de données ne sont pas modifiés par l'appel de ces commandes.

LOGO:

Le logo du Cerfacs peut être inséré en haut à droite du graphique par le lancement de la commande "go logo_cerfacs".

LABELS:

Les graphiques apparaissent sans label, sans axes et sans titres.

TITRES:

Un titre peut être ajouté après exécution d'une commande de visualisation par le lancement du script "titre.jnl".

CONTINENTS:

Pour les visualisations en projection non polaires, les contours des continents peuvent être ajoutés par le lancement de la commande "go land" (ou "go land_east" pour la grille Orca).

Des continents grisés peuvent être ajoutés par le lancement de la commande "fland.jnl".

ZOOM:

Pour zoomer, utiliser les options /xlim=longitude_min:longitude_max et /ylim=latitude_min:latitude_max dans l'argument "Commande du plot" (ou délimiter des aires géographiques par défaut: "set region"). En règle générale, pour les 3 types de grille, ne pas chercher à fixer des limites géographiques avec les options [I=...,J=...].

Pour zoomer sur des régions à cheval sur le méridien de Greenwich, sur une grille AT42, vérifier que la variable longitude (**ID**) de votre fichier de données comporte bien l'attribut "modulo".

1.4.3.2 Concernant les scripts de type *_polar.jnl

DROITS D'ECRITURE:

Ces scripts de visualisation en projection polaire utilisent des scripts de base "Ferret" polar_2d.jnl ou polar_vector.jnl. Ces scripts sauvent des fichiers de données contenant les nouvelles coordonnées en projection polaire (polar_grid.dat ou polar_[UV]_points.dat). Pour cela, il est nécessaire de se trouver

dans un répertoire où sont ouverts les droits en écriture lors du lancement de Ferret. A noter que ces fichiers de coordonnées sont ré-utilisables (l'utilisateur peut se passer de l'exécution du script SIBYLLE et lancer directement la commande polar_2d.jnl ou polar_vector.jnl). Cette remarque s'applique également aux scripts d'interpolation, de création de fond de carte, de calcul d'intégrale ou d'analyses en ondelettes.

CONTINENTS, CONTOUR DES GRILLES EN PROJECTION POLAIRE:

Sur le graphe en projection polaire, on pourra superposer des contours de continent (polar_land.jnl), des continents grisés (polar_fland.jnl) ou des grilles de latitude - longitude (polar_grid_fancy.jnl). Pour la syntaxe de ces scripts, voir leur entête dans le répertoire\$FER_GO (voir le script exferret).

1.4.3.3 *Concernant les scripts de type *_verti.jnl*

DISTANCE VERTICALE:

La distance verticale est fixée à 7000Km. Pour la changer, changer la valeur "7000" à la ligne

```
go mp_vertical_at42 $3"0" $2"90" 7000
```

ou

```
go mp_vertical_orca $3"0" $2"90" 7000
```

1.4.3.4 *Concernant les scripts de type *_ortho.jnl ou *_verti.jnl*

CONTINENTS EN PROJECTION ORTHO OU VERTICALE:

L'ajout de lignes de grilles ou de continents n'est pas possible avec les scripts standard "graticule.jnl" et "land.jnl" ou "fland.jnl". Pour masquer les continents, utiliser l'argument "Masque" de votre script SIBYLLE. Essayer aussi le script "mp_land.jnl".

1.4.3.5 *Concernant les scripts de type *_orca.jnl*

MASQUES:

Pour masquer un champ, il n'existe pas d'argument spécifique. La variable TORC_MSK est un masque terre qui peut être appliqué à un champ ORCA ne contenant pas de masque (fichier de forçage, résultat d'interpolation, grille Levitus ...). Pour cela, on visualisera le champ "CHAMP+TORC_MSK[d=masks]".

ZOOM:

Les zooms à cheval sur la ligne de recouvrement Ouest-Est (océan Indien) ne sont pas possible à l'aide de ce script (voir projection de la grille ORCA sur une grille régulière, scripts en préparation).

1.4.3.6 Concernant les scripts de visualisation de grilles Arpege:

DIMENSION TEMPORELLE:

Pour la grille régulière AT42: Penser à fixer la dimension temporelle pour les fichiers contenant plusieurs échéances

ex: go plot_at42 PLZOW200[L=1]
ou bien go plot_at42 PLZOW200 fill/L=1

Pour la grille réduite BT42: Ne pas chercher à passer l'option [L=...] avec le nom du champ, ni l'option [L=@shf:...]. Cet inconvénient disparaît à l'utilisation du script d'interpolation "interp_bt42.jnl".