

9.11. RÉSUMÉ DES COMMANDES RCL

9.11.1 Liste fonctionnelle des commandes RCL

Cette liste est celle obtenue par la commande **RCL_menu**, simplement remise en forme.

• *Software Informations :*

RCL_menu : return this list in ASCII format.
 RCL_list : return list of all RCL commands: create RCL_list.txt
 RCL_version : return current RCL version used, ex: RCL_V1p8
 RCL_make_minidoc : create file mini_doc.txt
 RCL_compare_versions : compare two different versions of RCL package
 RCL_compare_versions_long : same as previous but more deeper

• *Tools to handle RFF files (general)*

RCL_check_rff : check content of a RFF file
 RCL_check_rff_dir : check of all RFF files of a given directory
 RCL_info_rff : return main info of given RFF file
 RCL_info_rff_dir : return main info of all RFF files of a given directory
 RCL_clean_rff : clean content of a RFF file
 RCL_copy_rff : copy a RRF file into another, name updated
 RCL_diff_rff : compute difference between 2 RFF files
 RCL_move_rff : move a RFF file with update of file name inside
 RCL_reduce_time_rff : reduce the time period of a RFF file

• *Generic commands*

RCL_list_block_WF : Return the list of all blocks in a WF file
 RCL_waveform_to_vectime : Convertie un RFF de classe WaveForm vers un classe Vectime
 RCL_vectime_to_spectro : Produce a spectrogram from a VecTime file
 RCL_spectro_to_polar : Produce a spectrogram from a VecTime file

• *Access to local CLUSTER/STAFF-SC L1 data base*

RCL_get_data_CLUSTER_VTL1 : get entire day of CLUSTER/STA VTL1
 RCL_get_data_CLUSTER_VTL2 : get entire day of CLUSTER/STA VTL2
 RCL_get_data_CLUSTER_WFL1_forVTL1 : get entire day of CLUSTER/STA WFL1 + epsi before
 RCL_get_data_CLUSTER_VTL1_forVTL2 : get entire day of CLUSTER/STA VTL1 + epsi before/after
 RCL_get_data_CLUSTER_VTL1_forSPL2 : get entire day of CLUSTER/STA VTL1 + epsi before

• *Access to local CLUSTER/Orbit data base*

RCL_get_data_CLUSTER_POS : read .orb from local database and get 4 S/C positions/velocity
 RCL_get_data_CLUSTER_GEOM : read .orb from local database and get tetrahedron geometry
 RCL_give_CLUSTER_ORB_period : give list of orbit duration for all the mission

• *Access to local CLUSTER/FGM L2 data base*

RCL_get_data_CLUSTER_FGM : read the day file.cef in the local database and create a .bav
 RCL_fgm_cef_to_rff : convert a fgm.cef file into a fgm_VTL2.rff file
 RCL_fgm_cef_to_bav : convert a fgm.cef file into a fgm.bav file
 RCL_fgm_cef_gse_to_sr2 : convert a fgm.cef file with data in GSE into SR2 system

• *Access to CSA/FGM L2 data base*

RCL_download_data_CLUFGM_oneday : download cef data from CSA (SPIN, 5VPS or FULL)
 RCL_download_data_CLUFGM_onemonth : download cef data from CSA with tree creation
 RCL_download_data_CLUFGM_onemonth_nolog : same without log file
 RCL_download_data_CLUFGM_oneyear : download cef data from CSA with tree creation

• *CLUSTER/STAFF-SC processing*

RCL_vectime_L1_to_spectro_L2 : Calibrate a L1 STAFF-SC RFF and produce spectrogram
 RCL_vectime_calibration_CLUSTA : Calibrate a L1 STAFF-SC RFF VecTime file
 RCL_vectime_L1_to_cef : convert a RFF VTL1 into a cef DWF file
 RCL_vectime_L2_to_cef : convert a RFF VTL2 into a cef CWF file
 RCL_spectro_L2_to_cef : convert a SPL2.rff into a CS.cef
 RCL_give_spin_dir : give spin direction from VTL1 data base

• *Mass production for STAFF-SC :*

RCL_product_VTL1_oneday : product VTL1.rff file 1 day 1 sat
 RCL_product_VTL1_onemonth_nolog : product VTL1.rff files 1 month 4 sat
 RCL_product_VTL1_onemonth : product VTL1.rff files 1 month 4 sat + log_file
 RCL_product_VTL1_oneyear : product VTL1.rff files 1 year 4 sat
 RCL_product_VTL1_daily : product VTL1.rff file 1 day 1 sat without Tcor

RCL_product_VTL2_oneday : product VTL2.rff files 1 day 1 sat
 RCL_product_VTL2_onemonth_nolog : product VTL2.rff files 1 month 4 sat
 RCL_product_VTL2_onemonth : product VTL2.rff files 1 month 4 sat + log_file
 RCL_product_VTL2_onemonth_onesat : product VTL2.rff files 1 month 1 sat + log_file

RCL_product_SPL2_oneday : product SPL2.rff files 1 day 1 sat
 RCL_product_SPL2_onemonth_nolog : product SPL2.rff files 1 month 4 sat
 RCL_product_SPL2_onemonth : product SPL2.rff files 1 month 4 sat + log_file
 RCL_product_SPL2_oneyear : product SPL2.rff files 1 year 4 sat
 RCL_product_SPL2_oneday_fordaily : product SPL2.rff files 1 day 1 sat without Tcor
 RCL_product_SPL2_daily : product SPL2.rff files 1 day 4 sat 2 modes

RCL_product_DWF_oneday : product DWF.cef files 1 day 1 sat
 RCL_product_DWF_onemonth_nolog : product DWF.cef files 1 month 4 sat
 RCL_product_DWF_onemonth : product DWF.cef files 1 month 4 sat + log_file
 RCL_product_DWF_oneyear : product DWF.cef files 1 year 4 sat

RCL_product_CWF_oneday : product CWF.cef file 1 day 1 sat
 RCL_product_CWF_onemonth_nolog : product CWF.cef files 1 month 4 sat
 RCL_product_CWF_onemonth : product CWF.cef files 1 month 4 sat + log_file
 RCL_product_CWF_oneyear : product CWF.cef files 1 year 4 sat

RCL_product_CS_oneday : product CS.cef files 1 day 1 sat
 RCL_product_CS_onemonth_nolog : product CS.cef files 1 month 4 sat
 RCL_product_CS_onemonth : product CS.cef files 1 month 4 sat + log_file
 RCL_product_CS_oneyear : product CS.cef files 1 year 4 sat

RCL_product_visu_SPL2_oneday : product SPL2.ps files 1 day 4 sat 4Bz
 RCL_product_visu_SPL2_onemonth_nolog : product SPL2.ps files 1 month 4 sat 4Bz
 RCL_product_visu_SPL2_onemonth : product SPL2.ps files 1 month 4 sat 4Bz + log_file
 RCL_product_visu_SPL2_daily : product SPL2.ps files 1 day 4 sat 4Bz
 RCL_product_PNG_LowRes : product PNG_LowRes files from PS data base

• **Mass production for orbit visualization :**

RCL_product_visu_orbit_oneday : product orbit graphical files 1 day 4 sat
 RCL_product_visu_orbit_onemonth_nolog : product orbit graphical files 1 month 4 sat
 RCL_product_visu_orbit_onemonth : same with log file
 RCL_product_visu_orbit_oneyear : product orbit graphical files 1 year 4 sat

• **Time Utilities :**

RCL_current_date : return date as 2012-08-28 14:08:16 day 241 Julsec 1346155696
 RCL_nday_of_month : return number of day in a month
 RCL_next_day : return date of next day, as '20070925'
 RCL_previous_day : return date one day before, as '20070925'
 RCL_decode_datiso : for date as 2001-09-23T09:17:03.000Z return 2001 09 23 09 17 03
 RCL_decode_datim : for date as 20010923_091703 return 2001 09 23 09 17 03
 RCL_encode_datiso : return ISO_date from year month day hour min sec
 RCL_list_days_of_month : return a list of day for a given year/month as 01 02 03 30

• **Visualization tools :**

RCL_visu_spectro : visualization of a Spectrogram RFF file
 RCL_visu_spectro_3H : visualization of a Spectrogram RFF file (8 x 3H files)
 RCL_visu_spectro_4Bz : visualization of 4 Spectrogram RFF file Bz only
 RCL_visu_spectro_4Bz_3H : visualization of 4 Spectrogram RFF file Bz only (8 x 3H files)
 RCL_visu_ave_spectrum : visualization of an average spectrum from Spectrogram file
 RCL_visu_vectime : visualization of a VecTime RFF file
 RCL_visu_vectime_widget : launch a widget for using visu_vectime
 RCL_visu_polar : visualization of a copolar.resu file
 RCL_visu_CLUPOS : visualization of a clupos.resu file
 RCL_visu_CLUGEOM : visualization of a clugeom.resu file

• **Graphics Utilities :**

RCL_ps_to_pdf : create PDF file with or without image compression
 RCL_ps_to_png : create PNG file for a given resolution (dpi, 300=good) 16M colors
 RCL_ps_to_png_256 : create PNG file for a given resolution (dpi, 300=good) 256 colors

• **System & Directories properties :**

RCL_system_info : return system information (host, system, machine, OS etc.)
 RCL_dir_size : give directory size (octets or Mo)
 RCL_dir_size_tree : give directory size (octets or Mo) for all the tree
 RCL_dir_size_pretty_tree : same with pretty presentation
 RCL_dir_properties : give directory size (octets or Mo), number of files /directories
 RCL_dir_properties_tree : give directory properties for all the tree
 RCL_dir_properties_pretty_tree : same with pretty presentation
 RCL_search_duplicates : search and list duplicate files in a tree

Table 71: Liste fonctionnelle des commandes RCL (3 pages)

9.11.2 Liste alphabétique des commandes avec arguments

Usage and list of arguments can be given by using help option of each command. Table 72 hereafter give alphabetic list of all RCL commands.

RCL_check_rff_dir RCL_check_rff RCL_clean_rff RCL_compare_versions RCL_compare_versions_long RCL_copy_rff	toto toto.rff toto.rff RCL_previous_dir RCL_previous_dir toto1.rff toto2.rff
RCL_current_date RCL_decode_datiso RCL_decode_datim RCL_diff_rff RCL_dir_properties RCL_dir_properties_tree RCL_dir_properties_pretty_tree RCL_dir_size RCL_dir_size_tree RCL_dir_size_pretty_tree RCL_download_data_CLUFGM_oneday RCL_download_data_CLUFGM_onemonth RCL_download_data_CLUFGM_onemonth_nolog RCL_download_data_CLUFGM_oneyear RCL_encode_datiso	<i>/ No arguments</i> Datiso as 2001-09-23T09:17:03.000Z Date_time as 20010923_091703 toto1.rff toto2.rff <i>/Optional</i> : Mo, DIR, DIR Mo <i>/Optional</i> : Mo, DIR, DIR Mo <i>/Optional</i> : Mo, DIR, DIR Mo <i>/Optional</i> : Mo, DIR, DIR Mo <i>/Optional</i> : Mo, DIR, DIR Mo <i>/Optional</i> : Mo, DIR, DIR Mo SatNum year month day Mode year month Mode year month Mode year Mode year month day hour min sec
RCL_fgm_cef_to_bav RCL_fgm_cef_to_rff	toto.cef toto.cef
RCL_get_data_CLUFGM RCL_get_data_CLUGEOM RCL_get_data_CLUPOS RCL_get_data_CLUSTA_VTL1 RCL_get_data_CLUSTA_VTL2 RCL_get_data_CLUSTA_VTL1_forSPL2 RCL_get_data_CLUSTA_VTL1_forVTL2 RCL_get_data_CLUSTA_WFL1_forVTL1 RCL_give_CLUORB_period RCL_give_spin_dir	SatNum year month day Mode year month day hh mm ss duration dt coord year month day hh mm ss duration dt coord SatNum year month day BitRate SatNum year month day BitRate Coord SatNum year month day BitRate SatNum year month day BitRate SatNum year month day BitRate <i>/ No arguments</i> SatNum year month day
RCL_info_rff RCL_info_rff_dir	toto.rff option toto
RCL_list RCL_list_block_WF RCL_list_days_of_month	<i>/ No arguments</i> toto.rff year month
RCL_make_minidoc RCL_menu RCL_move_rff RCL_nday_of_month RCL_next_day RCL_previous_day	<i>/ No arguments</i> <i>/ No arguments</i> toto1.rff toto2.rff year month date date

RCL_product_CS_oneday RCL_product_CS_onemonth RCL_product_CS_onemonth_nolog RCL_product_CS_oneyear	SatNum year month day BitRate Coord year month BitRate Coord year month Coord year BitRate Coord
RCL_product_CWF_oneday RCL_product_CWF_onemonth RCL_product_CWF_onemonth_nolog RCL_product_CWF_oneyear	SatNum year month day Coord year month Coord year month Coord year Coord
RCL_product_DWF_oneday RCL_product_DWF_onemonth_nolog RCL_product_DWF_onemonth RCL_product_DWF_oneyear	SatNum year month day BitRate year month BitRate year month BitRate year BitRate
RCL_product_SPL2_daily RCL_product_SPL2_oneday RCL_product_SPL2_oneday_fordaily RCL_product_SPL2_onemonth RCL_product_SPL2_onemonth_nolog RCL_product_SPL2_oneyear	year month day BitRate Coord SatNum year month day BitRate Coord SatNum year month day BitRate Coord year month BitRate Coord year month BitRate Coord year BitRate Coord
RCL_product_visu_orbit_oneday RCL_product_visu_orbit_onemonth RCL_product_visu_orbit_onemonth_nolog RCL_product_visu_orbit_oneyear	year month day year month year month year
RCL_product_visu_SPL2_daily RCL_product_visu_SPL2_oneday RCL_product_visu_SPL2_onemonth RCL_product_visu_SPL2_onemonth_nolog	year month day BitRate Coord year month day BitRate year month BitRate year month BitRate Coord
RCL_product_VTL1_daily RCL_product_VTL1_oneday RCL_product_VTL1_onemonth RCL_product_VTL1_onemonth_nolog RCL_product_VTL1_oneyear	year month day BitRate SatNum year month day BitRate year month BitRate year month BitRate year BitRate
RCL_product_VTL2_oneday RCL_product_VTL2_onemonth RCL_product_VTL2_onemonth_nolog RCL_product_VTL2_onemonth_onesat	SatNum year month day BitRate Coord year month BitRate Coord year month BitRate Coord year month BitRate Coord
RCL_ps_to_pdf.sh RCL_ps_to_png.sh	toto.ps toto.ps 80
RCL_reduce_time_rff	toto1_VT.rff toto2_VT.rff datiso1 datiso2
RCL_search_duplicates RCL_spectro_L2_to_cef RCL_spectro_to_polar RCL_system_info	mode (I or B) toto_VT.rff toto_VT.rff <i>/ No arguments</i>
RCL_vectime_calibration_CLUSTERA RCL_vectime_L1_to_cef RCL_vectime_L1_to_spectro_L2 RCL_vectime_L2_to_cef RCL_vectime_to_spectro	VTL1.rff VTL2.rff Fdet Fc F1 F2 Step N-Kern N_shift Apod toto_VT.rff VTL1.rff SPL2.rff Fdet Fc F1 F2 Step N-Kern N_shift Apod toto_VTL2.rff VT.rff SP.rff N-Kern N_shift Apod

RCL_version	<i>/ No arguments</i>
RCL_visu_ave_spectrum	SP.rff datiso1 datiso2 f1 f2 Pmin Pmax XY/LR info_orb info_fgm
RCL_visu_CLUGEOM	<i>/ No arguments</i>
RCL_visu_CLUPOS	<i>/ No arguments</i>
RCL_visu_polar	copolar_file threshold
RCL_visu_spectro	SP.rff datiso1 datiso2 f1 f2 pmin pmax
RCL_visu_spectro_3H	SP.rff f1 f2 puimin puimax f_int1 f_int2
RCL_visu_spectro_4Bz	SP1.rff SP2.rff SP3.rff SP4.rff datiso1 datiso2 \
RCL_visu_spectro_4Bz_3H	SP1.rff SP2.rff SP3.rff SP4.rff f1 f2 \
RCL_visu_vectime	toto_VT.rff
RCL_visu_vectime_widget	<i>/ No arguments</i>
RCL_waveform_to_vectime	toto_WF.rff toto_VT.rff

Table 72: *Liste alphabétique des commandes RCL avec arguments (3 pages)*

9.12. QUELQUES MOTS SUR LES ROPROC

9.12.1 *Historique des Roproc et lien avec les RCL*

Les procédures Roproc (Les « Robert's processing ») ont été développées au CETP dans les années 2000 pour l'exploitation scientifique des données CLUSTER, et ont été en constante évolution durant les premières années d'exploitation. Elles s'appuient pour STAFF sur la base en ascii des données NI décommutées, pour FGM sur la base des PPDB (fichiers ascii au format standard FGM, dits fichiers « bav » comme « B average »), et sont aussi valables pour la bases des fichiers FGM à haute résolution mais au même format, dits « bhr » comme « B Hight resolution ». Pour les fichiers d'orbite, les procédures s'appuient sur la base des fichiers d'orbite (extension .orb) produits lors du traitement de routine des données STAFF.

Ces procédures produisent des fichiers de résultats en ascii, auto-documentés, et de format aisément compréhensible même en l'absence de programme de lecture : ils sont repris par des procédures de visualisation créant des fichiers PostScript. Les programmes de lecture de ces fichiers de résultats sont disponibles en Fortran 90 ou en IDL.

Les procédures pour STAFF-SC sont largement dérivées des anciennes procédures de traitement des données GEOS, remises à niveau à l'époque pour l'archivage de ces données au CDDP, et étendues à 4 satellites. Les procédures pour FGM et pour les orbites ont été développées dans le même esprit, en utilisant au maximum les bibliothèques existantes, éventuellement étendues. Elles utilisent notamment la bibliothèque de changement de coordonnées Rocotlib [voir 8, Robert, 2003].

Beaucoup de codes de traitement originaux étaient en FORTRAN 77, et ont été reconvertis en F90. Les nouveaux codes ont été écrits en F90. Toutes les visualisations sont en IDL. Les visualisations s'appuient sur un standard développé pour CLUSTER, et sont regroupées en bibliothèques, avec des modules spécialisés, ce qui facilite grandement la création d'éventuelles nouvelles visualisations.

Les procédures Roproc sont livrés sous la forme d'une arborescence de fichiers exécutables, et peuvent être installées sur une machine Linux ou Windows par simple recopie de l'arborescence correspondante du logiciel : Un pack IDL de démonstration est inclus pour les plateformes Sun, Linux, Windows et Mac PPC, ce qui fait qu'une licence IDL n'est pas nécessaire pour travailler et pour produire des graphiques PostScript, PDF ou PNG.

Les Roproc ont servi d'inspiration au projet RCL. En effet plusieurs programmes, ou éléments de programmes, ont été récupérée pour fabriquer les RCL. Néanmoins la programmation en F77 convertie en F90 des Roproc n'étaient pas très propres ni très performante pour un projet devant traiter en production 10 à 15 ans de données. Les codes F90 des RCL ont donc été soigneusement révisés, voire réécrits à la norme F90, en soignant les codes d'erreurs et les fins anormales, afin d'obtenir des codes fiables et performants.

La courte documentation des Roproc données ci-après peut être utile si on souhaite développer d'autres commandes RCL. Néanmoins une adaptation aux bibliothèques RCL décrite au § 7.3 peut être nécessaire.

Les informations ci-dessous sont directement issue du répertoire « doc » du pack Roproc 4.3. Une documentation des Roproc est prévue, mais elle n'existe pas encore à l'heure ou est écrit ce document. A l'origine, le projet RCL devait englober la totalité des Roproc, rendant ce dernier obsolète. Mais le temps a manqué pour mener ce projet à terme, et les deux produits existent aujourd'hui simultanément.

9.12.2 Introduction aux Roprocs

```

ROPROC SHORT DOCUMENTATION

Creation date : 2010-12-14 18:01:16

Roproc version= Roproc_4p3

HOST NAME= cactus
HOST TYPE= x86_64-linux
OS TYPE= linux
MACH TYPE= x86_64

Author : Patrick ROBERT, 2000-2010, CNRS/CETP-LPP

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
%%              INTRODUCTION              %%
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

RoProc software is a set of procedures, or commands, allowing data processing
and visualisations of time series data. Some general info commands are available
(see Part II).

There is two mains categories of data:

- waveform data, where the sample rate is constant.
  Fast Fourier Transform, polarisation computation, etc. are available on
  this kind of data.

  data format are vector block data, time tagged. Example is STAFF experiment
  of the CLUSTER mission.

  The list of available Roproc Wave commands are given hereafter.

- vector data, where the sample rate can be varying, or could have data gaps

  data format are vector data, where each vector is time tagged.
  Example is FGM experiment of the CLUSTER mission

  In case of constant sampling rate, conversion with wave format is possible.
  The list of available Roproc Vector commands are given hereafter.

  There is also a set of Roproc Orbit commands, only available for CLUSTER
  mission, and a set of Roproc Magnetcs Tools, for magnetic field
  computaion (dipole, IGRF, Tsyganenko 87 until 04), conjugated points,
  fieldline computation, etc.

```

Table 73 : *Introduction aux Roprocs*

9.12.3 Liste thématique des Roprocs

LIST OF ROPROC COMMANDS, SCRIPTS AND MACROS

V 1.1 : P. Robert, CNRS/CETP, 2001 November 9
Last update: V 4.5 : P. Robert, CNRS/LPP, 2015 January 13

A) ROPROC COMMANDS

I) Roproc Info commands

```

rpc_menu      : give menu of all available commands (this file)
rpc_list      : give full alphabetic list of all rpc_commands
rpc_stat      : List of last used Roproc commands
rpc_version   : give version number of Roproc software

```

II) Roproc Wave commands

1) creating cowave.resu data files

```

rpc_cowave_rcs : Calibration of CLUSTER/STAFF-SC L1 data (by windows)
rpc_cowave_rce : Calibration of CLUSTER/EPW flat data *** disused
rpc_cowave_rds : Calibration of DSP/STAFF-SC N1p data *** disused

rpc_cowave_rcs4 : Run 4 times rpc_cowave_rcs for 4 spacecraft
rpc_cowave_rce4 : Run 4 times rpc_cowave_rce for 4 spacecraft *** disused

```

2) reading cowave.resu files

```

rpc_readwave : Read and check a cowave.resu file

```

3) filters : transform cowave.resu into another cowave.resu

```

rpc_wave_filter      : Filtering of a cowave.resu file
rpc_wave_to_mfa_rce  : Transform a cowave.resu file into mfa system *** disused
rpc_wave_to_mfa_rcs  : Transform a cowave.resu file into mfa system
rpc_wave_to_mfa_rds  : Transform a cowave.resu file into mfa system *** disused
rpc_wave_to_gse      : Transform a cowave.resu file into gse system
rpc_wave_to_gsm      : Transform a cowave.resu file into gsm system
rpc_wave_to_mva      : Transform a cowave.resu file into MVA system
rpc_wave_to_mat      : Transform a cowave.resu file via a given matrix

rpc_wave_filter4     : Run 4 times rpc_wave_filter for 4 S/C
rpc_wave_to_mfa_rce4 : Run 4 times rpc_wave_to_mfa_rce for 4 S/C
rpc_wave_to_mfa_rcs4 : Run 4 times rpc_wave_to_mfa_rcs for 4 S/C
rpc_wave_to_gse4     : Run 4 times rpc_wave_to_gse for 4 S/C
rpc_wave_to_gsm4     : Run 4 times rpc_wave_to_gsm for 4 S/C
rpc_wave_to_mva4     : Run 4 times rpc_wave_to_mva for 4 S/C
rpc_wave_to_mat4     : run 4 times rpc wave to mat for 4 S/C

```

4) level 3: transform cowave.resu into L3 product

```

rpc_copolar      : Computation of wave polarisation from cowave.resu data
rpc_cospectro    : Computation of spectrogram from cowave.resu data

rpc_copolar4     : Run 4 times rpc_copolar for 4 spacecraft
rpc_cospectro4   : Run 4 times rpc_cospectro for 4 spacecraft

```

Roproc_menu V4.5 Page 1/6

III) Roproc Vector commands

1) creating covector.resu data files

rpc_get_data_clufgm : Get FGM data and create the covector.resu file
 rpc_get_data_clufgm4 : Run 4 times rpc_get_data_clufgm for 4 spacecraft

2) reading covector.resu files

rpc_readvector : Read and check a readvector.resu file

3) filters : transform covector.resu into another covector.resu

rpc_vector_geo_to_gse : Change data coordinate system from GEO to GSE
 rpc_vector_gse_to_gsm : Change data coordinate system from GSE to GSM
 rpc_vector_gse_to_sma : Change data coordinate system from GSE to SMA
 rpc_vector_gse_to_sr2 : Change data coordinate system from GSE to SR2
 rpc_vector_to_mfa : Change data coordinate system from any to MFA
 rpc_vector_to_mva : Change data coordinate system from any to MVA
 rpc_vector_to_mat : Change data coordinate system with given matrix

rpc_vector_minus_line : Soustract a line at covector.resu data
 rpc_vector_minus_trend : Remove the tendency at covector.resu
 rpc_vector_minus_sine : compute and remove a sine signal from a file

rpc_vector_reduce_time : extract a time period from a file
 rpc_vector_subsample : subsample a series on n points by averaging
 rpc_vector_timeshift : Time shift of covector.resu data
 rpc_vector_to_model : Change a covector.resu FGM data by a mag. model

rpc_vector_minus_vector : Difference between two covector.resu files
 rpc_vector_cross_product : compute A X B for each vector of files A & B
 rpc_vector_average : Average several covector.resu files
 rpc_vector_alitime : Align the time of covector.int.resu on this of covector.ref.resu, result in covector.resu

rpc_vector_geo_to_gse4 : Run 4 times rpc_vector_geo_to_gse for 4 S/C
 rpc_vector_gse_to_gsm4 : Run 4 times rpc_vector_gse_to_gsm for 4 S/C
 rpc_vector_gse_to_sma4 : Run 4 times rpc_vector_gse_to_sma for 4 S/C
 rpc_vector_gse_to_sr24 : Run 4 times rpc_vector_gse_to_sr2 for 4 S/C
 rpc_vector_to_mfa4 : Run 4 times rpc_vector_to_mfa for 4 S/C
 rpc_vector_to_mva4 : Run 4 times rpc_vector_to_mva for 4 S/C
 rpc_vector_to_mat4 : Run 4 times rpc_vector_to_mat for 4 S/C

rpc_vector_minus_line4 : Run 4 times rpc_vector_minus_line for 4 S/C
 rpc_vector_minus_trend4 : Run 4 times rpc_vector_minus_trend for 4 S/C
 rpc_vector_reduce_time4 : Run 4 times rpc_vector_reduce_time for 4 S/C
 rpc_vector_subsample4 : Run 4 times rpc_vector_subsample for 4 S/C
 rpc_vector_timeshift4 : Run 4 times rpc_vector_timeshift for 4 S/C
 rpc_vector_minus_vector4 : Run 4 times rpc_vector_minus_vector for 4 S/C
 rpc_vector_to_model4 : Run 4 times rpc_vector_to_model for 4 S/C

5) level 3: transform covector.resu into L3 product

rpc_addvecpos : Add position of the satellite in covector.resu
 rpc_addvecpos4 : Run 4 times rpc_addvecpos for 4 spacecraft
 rpc_alivetime : Time synchronisation of 4 files covector.i.resu
 rpc_cocurldiv : Compute curl & div from 4 time-aligned addvecpos_N.resu

rpc_merge_alivector : Merge 4 time-aligned covector.N.resu files into a single one

IV) Roproc files conversion**1) conversion to vector format**

rpc_wave_to_vector : Convert a cowave.resu file into covector.resu
 rpc_wave_to_vector4 : Run 4 times rpc_wave_to_vector for 4 S/C
 rpc_fgm_bav_to_vector: Convert a fgm bav file into a covector.resu

2) conversion to wave format

rpc_vector_to_wave : Convert a covector.resu file into a cowave.resu
 rpc_vector_to_wave4 : Run 4 times rpc_vector_to_wave for 4 S/C

3) other data file conversions

rpc_vector_to_bav : Convert a covector.resu file into a fgm bav file
 rpc_fgm_cef_to_bav : Convert a fgm cef file into a fgm bav file
 rpc_vectime_to_covector : Convert RFF vectime files to covector format
 rpc_rff_reduce_time : reduce the time period of a RFF file (WF or VT class)

4) Image conversion

rpc_ps_to_pdf : Convert PostScript into a PDF file
 rpc_ps_to_png : Convert PostScript into a PNG file

V) Roproc Orbit commands

rpc_creatpos_rco : Creating list of Cluster positions
 rpc_listorbit_rco : Creating list of Cluster orbit over 24 hours
 rpc_get_param_orbi_cli : Computation of Cluster orbit parameters
 rpc_compute_trajectory : Compute elliptical trajectory of a Earth satellite
 rpc_test_keplerlib : Compute Earth orbit, to check routines
 rpc_co_cov_matrix : Compute covariance matrix for a given date
 rpc_co_discont : Computation of discontinuity velocity from time delays

VI) Roproc Magnetic tools

rpc_conjug : Computing the conjugate points by Tsyganenko model
 rpc_cotsyfield : Computation of Tsyganenko magnetic Field
 rpc_cofieldline : Computation of a field line model
 rpc_cofieldvector : Computing a field vector by Tsyganenko model

VII) Roproc Utilities**1) STAFF files**

rpc_datafiles : Print the list of the available L1 files
 rpc_libloc : List of all blocs of a L1 file
 rpc_exit_on_error : used only with arg \$? \$@ into a shell
 rpc_testgain : Test the gain of the antennas
 rpc_test_time : Test the time continuity in a L1 file
 rpc_give_spin_direction : Give rught asc. and decl. of spin axis for given date
 rpc_cal_CWSTA : Continuous calibration of a L1 STAFF-SC RFF Vectime file

4

2) Time utilities

```

rpc_caljul           : compute Julian day for a given date
rpc_current_date    : return current date and Julian sec.
rpc_today           : same, but pretty formatted
rpc_nday_of month   : give number of day of a given month

```

3) System utilities

```

rps_sys_info        : give system info (host, processor, operating system etc.)
rpc_dir_size        : give size of a given directory
rpc_dir_size_tree   : same for all the subtree
rpc_dir_properties  : give size & nb dir. & nb files in a given directory
rpc_dir_properties_tree : same for all the subtree

```

VIII) Roproc Visualization commands**1) wave visualization**

```

rpc_visuwave        : Visualization of cowave.resu waveforms
rpc_visuspectro     : Visualization of cospectro.resu spectrogram
rpc_visuspectro_4Bz : Visualization of 4 Bz Spectrogrammes
rpc_visuspectra     : Visualization of average spectrum from cospectro.resu
rpc_visupower       : Visualization of integrated power from cospectro.resu
rpc_visupolar       : Visualization of copolar.resu file

rpc_visuwave4       : Run 4 times rpc_visuwave for 4 spacecraft
rpc_visuspectro4    : Run 4 times rpc_visuspectro for 4 spacecraft
rpc_visuspectra4    : Run 4 times rpc_visuspectra for 4 spacecraft
rpc_visupower4      : Run 4 times rpc_visupower for 4 spacecraft
rpc_visupolar4      : Run 4 times rpc_visupolar for 4 spacecraft

```

2) vector visualization

```

rpc_visuvector      : Visualization of covector.resu file
rpc_visuvector_4sat : Visualization of 4 covector.resu files
rpc_visuaddvecpos   : Visualization of Cluster orbit over 3 planes
rpc_visucocurldiv   : Visualization of cocurldiv.resu file

rpc_visuvector4     : Run 4 times rpc_visuvector for 4 spacecraft
rpc_visuaddvecpos4  : Run 4 times rpc_visuaddvecpos for 4 spacecraft

rpc_visuvector_3D   : Visualization 3D of a covector.resu file

```

3) orbit visualization

```

rpc_visuorbit      : Visualization of listorbit files
rpc_visusatpos     : Visualization of Cluster orbit over 3 planes +3D persp.

```

Roproc Menu V4.5 - Page 4/6

IX) Some more about rpc commands

1) On line help

Usage and list of arguments can be given by using help option.

Example:

```
rpc_vector_to_mva help
```

2) rpc_cowave_xxx

Same sample rate is used for CLUSTER/STAFF-SC, CLUSTER/EFW, DSP/STAFF-SC. So, the nbp parameter of cowave procedure correspond to the same time duration for these 3 experiments. Correspondence between nbp and time duration is given below. "*" is a recommended value to produce spectrograms.

Nb. of pts	Duration in NBR (25 Hz)	Duration in HBR (450 Hz)
128	5.12 s *	0.2844 s
256	10.24 s *	0.5689 s
512	20.48 s *	1.138 s
1024	40.96 s *	2.275 s
2048	81.92 s or 1 m 22 s	4.551 s
4096	163.8 s or 2 m 44 s	9.102 s *
8192	327.7 s or 5 m 28 s	18.20 s *
16384	655.4 s or 10 m 55 s	36.40 s *
32768	1310.7 s or 21 m 51 s	72.82 s or 1 m 13 s
65536	2621.4 s or 43 m 41 s	145.6 s or 2 m 26 s

3) rpc_mave_to_mfa

rpc_wave_to_mfa_rcs : use CLUSTER/FGM PPD data to define the MFA system. Filtering on PPD data is done, and STAFF data are processed.

rpc_wave_to_mfa_rcs4 : Run 4 times rpc_wave_to_mfa_rcs for 4 S/C

rpc_mave_to_mfa_rce : Same as above, but EFW data are processed on the X and Y components of the SR2 system; 3 components in MFA are computed, with assumption of $E//=0$ (no EZ MFA component). *** disused

6

B) ROPROC SCRIPTS

Roproc scripts are shell scripts, running without arguments. Examples available in sh directories are following:

```

rps_polar_CLUSTER_ex1.sh  rps_polar_CLUSTER_ex2.sh
rps_CLUSTER_ex1.sh       rps_CLUSTER_ex2.sh
rps_polar_CLUSTER_1sat.sh rps_polar_CLUSTER_4sat.sh *recommended
rps_CLUSTERAFGM_ex1.sh   rps_CLUSTERAFGM_ex2.sh   rps_CLUSTERAFGM_ex3.sh
rps_CLUORB_ex.sh
rps_crefigdoc.sh

```

The user can copy these script files (from sh directory) into his directory, modify it, and create its own script files.

C) ROPROC MACROS

Roproc macros are Roproc commands built from a sequence of several Roproc commands. This enables automation of certain processing by the use of a single command.

Examples available in sh directories are following.

Note: To avoid splash visu, strike: `export R_VISU_SPLASH=no`

```

rpm_spectro_4sat_CLUSTER : produce spectrograms 4 S/C for CLUSTER/STAFF
ex: rpm_spectro_4sat_CLUSTER 2001 09 23 09 20 00 40. 512 0 0.1 0. 0. 5 NBR

rpm_polar_4sat_CLUSTER : produce polar plots for 4 S/C for CLUSTER/STAFF
ex: rpm_polar_4sat_CLUSTER 2001 09 23 09 20 00 40. 512 0 0.1 0. 0. 5 NBR

rpm_visu_4sat_CLUSTER : Produce 4 S/C plots of CLUSTER/FGM data
ex: rpm_visu_4sat_CLUSTER 2001 01 26 11 30 00 60. SPIN

rpm_curl_div_CLUSTER : Produce plots of curlB & divB from CLUSTER/FGM data
ex: rpm_curl_div_CLUSTER 2001 01 26 11 30 00 60. SPIN

```

The user can copy these macros (from sh directory) into his directory, modify it, and create its own macro commands.

Any problem ? patrick.roberty@ipp.polytechnique.fr

Roproc Menu V4.5 - Page 6/6

Table 74: *Liste thématique des Roproc (6 pages)*

9.12.4 Examples of arguments of Roproc commands

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
%%          EXAMPLES OF ARGUMENTS OF ROPROC COMMANDS          %%
%%          (Alphabetic order)                                %%
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

rpc_addvecpos CLUSTER
rpc_addvecpos4 CLUSTER
rpc_alivetime 6000
rpc_caljul 2002 09 23
rpc_cocurldiv
rpc_cofieldline 1996 12 15 21 00 00 t87s 6 1 0 0 0 0 0 0 0 0 \
gsm c RE -23.04 6.04 -1.4
rpc_cofieldvector 1996 12 15 21 00 00 t87s 6 1 0 0 0 0 0 0 0 0 \
gsm c RE -23.04 6.04 -1.4
rpc_compute_trajectory 124252.735203443d0 26321.1522350593d0 \
2001-02-24T15:17:23.000Z \
-0.9493840 0.3011743 0.0892410 \
0.0820478 -0.0364760 0.9959607 5.0
rpc_conjug 1996 12 15 21 00 00 t87s 6 1 0 0 0 0 0 0 0 0 \
gsm c RE -23.04 6.04 -1.4
rpc_copolar
rpc_copolar4
rpc_cospectro 1. 0. 0 0
rpc_cospectro4 0.5 0. 1 0
rpc_cotsyfield 1996 12 15 21 00 00 t89c 4 1 0 0 0 0 0 0 0 0 \
gsm c -23.04 6.04 -1.4
rpc_cowave_rce $RCE_DATA/efw4_20020909_SC.flat \
2002 09 09 4 41 0 0.5 4096 0.25 1 NBR
rpc_cowave_rce4 $RCE_DATA/efw4_20020909_SC.flat \
2002 09 09 4 41 0 0.5 4096 0.25 1 NBR
rpc_cowave_rcs 3 2001 09 23 09 20 00 40.1 512 0. 0.1 0. 0. 5 NBR
rpc_cowave_rcs4 2001 09 23 09 20 00 40.1 512 0 0.1 0. 0. 5 NBR
rpc_cowave_rds 1 2004 02 19 19 10 00 30. 512 0. 0.2 1.5 0. 5 NBR
rpc_cresatpos_rco 2001 09 23 09 00 00 360.0 5.0 gse
rpc_datafiles
rpc_get_data_clufgm 3 2001 09 23 09 20 00 40.1 bav
rpc_get_data_clufgm4 2001 09 23 09 20 00 40.1 bav
rpc_get_param_orbi_clu 2001 09 23 12 00 00
rpc_libloc $RCS_DATA/2001_09/010923_sat4_sc.n1p CLUSTER 4 NBR
rpc_list
rpc_listorbit_rco 2001 09 23 15 gse l
rpc_menu
results are in standard output
rpc_merge_alivector
rpc_readvector
rpc_readwave
rpc_stat 20
rpc_test_gain CLUSTER 1
rpc_test_keplerlib
rpc_test_time 3 2001 09 23 0 100 CLUSTER NBR
rpc_vector_alitime 6000
rpc_vector_average covector_1.resu covector_2.resu \
covector_3.resu covector_4.resu
rpc_vector_geo_to_gse
rpc_vector_geo_to_gse4
rpc_vector_gse_to_gsm
rpc_vector_gse_to_gsm4
rpc_vector_gse_to_sma
rpc_vector_gse_to_sma4
rpc_vector_gse_to_sr2 102.50 -63.90
rpc_vector_gse_to_sr24 102.50 -63.90
rpc_vector_minus_line -0.00833 10. 0. -22.5 0.01666 -8.
rpc_vector_minus_line4 -0.00833 10. 0. -22.5 0.01666 -8.

```

```

rpc_vector_minus_trend 100
rpc_vector_minus_trend4 100
rpc_vector_minus_vector vector_average.resu -Bmoy
rpc_vector_minus_vector4 bmoy.resu Bcm
rpc_vector_subsample 10
rpc_vector_subsample4 10
rpc_vector_timeshift 10.
rpc_vector_timeshift4 10.
rpc_vector_to_mat "MVA Bcm" \
  0.673415E+00  0.508084E+00 -0.536994E+00 \
  -0.470780E+00 -0.265316E+00 -0.841412E+00 \
  -0.569982E+00  0.819425E+00  0.605279E-01
rpc_vector_to_mat4 "MVA Bcm" \
  0.673415E+00  0.508084E+00 -0.536994E+00 \
  -0.470780E+00 -0.265316E+00 -0.841412E+00 \
  -0.569982E+00  0.819425E+00  0.605279E-01

rpc_vector_to_mfa
rpc_vector_to_mfa4
rpc_vector_to_model CLUSTER t87s 3 1 0 0 0 0 0 0 0 0
rpc_vector_to_model4 CLUSTER t87s 3 1 0 0 0 0 0 0 0 0
rpc_vector_to_mva
rpc_vector_to_mva4
rpc_vector_to_wave 512 0.25 25. 102.64 -63.88
rpc_vector_to_wave4 512 0.25 25. 102.64 -63.88
rpc_version
rpc_visuaddvecpos 2400. nT/RE 3
rpc_visuaddvecpos4 2400. nT/RE 3
rpc_visucocurldiv
rpc_visuorbit
rpc_visupolar -5.5
rpc_visupolar4 -5.5
rpc_visupower 0.5 0. 0. 0.
rpc_visupower4 0.5 0. 0. 0.
rpc_visusatpos
rpc_visuspectra 0. 0. 0. 0.
rpc_visuspectra4 0. 0. 0. 0.
rpc_visuspectro 0. 0. 0. 0.
rpc_visuspectro4 0. 0. 0. 0.
rpc_visuspectro 4Bz
rpc_visuvector 0. 0. 0. 0. 0. 0. 0. 0.
rpc_visuvector4 0. 0. 0. 0. 0. 0. 0. 0.
rpc_visuvector_4sat 0. 0. 0. 0. 0. 0. 0. 0.
rpc_visuwave 0 0. 0. 0. 0. 0. 0.
rpc_visuwave4 0 0. 0. 0. 0. 0. 0.
rpc_wave_filter 0.2 5.
rpc_wave_filter4 0.2 5.
rpc_wave_to_gse
rpc_wave_to_gse4
rpc_wave_to_gsm
rpc_wave_to_gsm4
rpc_wave_to_mat TPN_12.0  0.71279  -0.46706  -0.52324 \
  0.00000  0.74602  -0.66592 \
  0.70138  0.47466  0.53176
rpc_wave_to_mat4 TPN_12.0  0.71279  -0.46706  -0.52324 \
  0.00000  0.74602  -0.66592 \
  0.70138  0.47466  0.53176

rpc_wave_to_mfa_rce
rpc_wave_to_mfa_rce4
rpc_wave_to_mfa_rcs
rpc_wave_to_mfa_rcs4
rpc_wave_to_mfa_rds
rpc_wave_to_mva
rpc_wave_to_mva4
rpc_wave_to_vector
rpc_wave_to_vector4

```

END OF ROPROC SHORT DOCUMENTATION

Table 75: *Examples of arguments of Roproc commands (2 pages)*

10. LISTE DES ACRONYMES

STAFF : Spatio Temporal Analyse of Field Fluctuations

FGM : Flux Gate Magnetometer

HBR : High Bit Rate

NBR : Normal Bit Rate

RFF : Roproc File Format

RCL : Roproc Command Language

WFL1 : RFF Waveform file Level 1 (Uncalibrated)

VTL1 : RFF Vectime file Level 1 (Uncalibrated)

VTL2 : RFF Vectime file Level 2 (Calibrated)

SPL2 : RFF Spectrogram file Level 2 (Calibrated)

CAA : Cluster Active Archive

CSA : Cluster Science Archive

CEF : Cluster Exchange Format

DWF : Decommutated WaveForms

CWF : Calibrated WaveForms

CS : Complex (Calibrated) Spectra

FFT : Fast Fourier Transform

11. LISTE DES TABLES ET FIGURES

11.1. LISTE DES FIGURES

Figure 1 :	Chaîne de production des WFL1.rff. Les CD sont préalablement concaténés, ainsi que les SATT files et les HK satellites associés.	25
Figure 2 :	Chaîne de production des VTL1.rff. Les fichiers WFL1.rff du jour courant ET du jour précédent sont requis.	26
Figure 3 :	Chaîne de production des SPL2.rff. Les fichiers VTL1.rff du jour courant ET du jour suivant sont requis.	27
Figure 4 :	Chaîne de production des VTL2.rff. Les fichiers WFL1.rff du jour courant, du jour précédent ET du jour d'après sont requis.	28
Figure 5 :	Chaîne de production des DWF.cef. Celle-ci se résume à un simple filtre de changement de format, ne concernant que la partie méta data.	29
Figure 6 :	Chaîne de production des CS.cef. Celle-ci se résume à un simple filtre de changement de format, ne concernant que la partie méta data.	29
Figure 7 :	Chaîne de production des CWF.cef. Celle-ci se résume à un simple filtre de changement de format, ne concernant que la partie méta data.	30
Figure 8 :	Chaîne de production des images à mettre sur le web. Celles-ci servent aussi à la validation des données et des chaînes de traitement.	31
Figure 9 :	Chaîne de production des spectrogrammes pour la validation des CWF.	32
Figure 10 :	Schéma général des chaînes de production STAFF-SC.	33
Figure 11 :	Exemple de spectrogramme des données FGM produit à partir des commandes RCL.	87
Figure 12 :	Signal utile superposé à la sinusoïde de spin pour un search-coil en rotation.	124
Figure 13 :	apodisation en trapèze du signal pour la calibration d'un spectre.	127
Figure 14 :	Principe du "detrend".	133
Figure 15 :	représentation des fenêtres de pondérations disponibles.	137
Figure 16 :	Définition du « spin phase » dans le repère SR2 (extraite du DDID).	146
Figure 17 :	Position des antennes de STAFF dans le repère « Body Build » lié au satellite.	149
Figure 18 :	Definition of SR system.	150
Figure 19 :	Definition of SR2 system (Despun).	151
Figure 20 :	Definition of GEI system:	153
Figure 21 :	Definition of GSE system.	153
Figure 22 :	Definition of GSM system.	154
Figure 23 :	Definition of MFA system.	154
Figure 24 :	Mesure du champ perpendiculaire et de la phase dans le plan de spin.	155
Figure 25 :	Définition des repères pour un capteur en rotation.	157
Figure 26 :	Projection du vecteur J sur 3 vecteurs (P, Q, R) non orthogonaux.	161
Figure 27 :	Différence entre les composantes mathématique de B et les mesures.	162
Figure 28 :	Ordre des raies du spectre en sortie de FFT.	168
Figure 29 :	Spectre d'une sinusoïde pure dont la fréquence est un multiple de e f.	169
Figure 30 :	Spectre d'une sinusoïde pure dont la fréquence n'est pas un multiple de e f.	170
Figure 31 :	Transformée de Fourier du signal complexe $S_{xy} = x + iy$ associé à une onde plane monochromatique pour 3 exemples particuliers (Robert, 1971, [20]).	171
Figure 32 :	Passage du repère en rotation à un repère fixe par les composantes circulaires gauche et droite (Robert, 1971, [20]).	172
Figure 33 :	Exemple de visu_spectro à partir des VTL2.	185

Figure 34:	Exemple de visu_spectro à partir des VTL1	186
Figure 35 :	Exemple de visu_spectro_4Bz utilisée comme « Quick looks »	187
Figure 36:	Exemple de visu_spectra à partir des VTL1	188
Figure 37:	Exemple de visu_vectime sur un VTL1	189
Figure 38:	Exemple de visu_vectime sur un VTL2 en ISR2	190
Figure 39:	Exemple de visu_vectime sur un VTL2 en GSE	191
Figure 40:	Exemple de spectrogramme en repère MFA et en composantes circulaires, montrant la polarisation circulaire droite d'un whisler dans le plan perpendiculaire au champ magnétique DC local.....	192
Figure 41:	Spectre moyen du spectrogramme précédent, montrant la nette prépondérance du mode droit (vert).....	193
Figure 42:	Exemple de visualisation du résultat du programme de calcul de polarisation.....	194
Figure 43:	Exemple d'onde pseudo monochromatique linéaire dans le plan perpendiculaire au champ magnétique DC local	195
Figure 44:	Autre exemple de résultat du calcul de polarisation.....	196
Figure 45:	Exemple de visu_CLUPOS sur 3 heures en GSE	197
Figure 46:	Exemple de visu_CLUPOS sur 3 heures en GSM.....	198
Figure 47:	Exemple de visu_CLUPOS sur une orbite complète en GSE	199
Figure 48:	Exemple de visu_CLUPOS sur une orbite complète en GSM.....	200
Figure 49:	Exemple de visu_CLUGEOM, partie « positions et vitesses »	201
Figure 50:	Exemple de visu_CLUGEOM, partie « géometrie du tétraèdre ».....	202

11.2. LISTE DES TABLES

Table 1 :	Exemple de script RCL pour calculer et visualiser un spectrogramme de données calibrées.....	16
Table 2:	Exemple de passage d'un format WF (WaveForm) à un format VT (VecTime) par l'application RCF_waveform_to_vectime.....	18
Table 3 :	Nomenclature des noms de fichiers RFF	22
Table 4:	Nomenclature des noms de fichiers RFF	22
Table 5 :	Arborescence des fichiers RFFet PNG.....	23
Table 6:	Arborescence des fichiers CEF.....	23
Table 7:	Arborescence des fichiers CEF.....	24
Table 8:	Cluster WEC (DWP) exact sampling frequency derived from the DWP clock	36
Table 9 :	Valeurs des paramètres de la commande RCL_product_VTL2_oneday	36
Table 10:	Valeurs des paramètres de la commande RCL_product_SPL2_oneday.....	37
Table 11:	Valeurs des paramètres de la commande RCL_product_visu_SPL2_oneday	37
Table 12:	Taille et nombre de fichiers de l'arborescence des fichiers WFL1 « zippés »	40
Table 13:	Taille et nombre de fichiers de l'arborescence des fichiers VTL1	41
Table 14:	Taille et nombre de fichiers de l'arborescence des fichiers VTL2 « zippés »	42
Table 15:	Taille et nombre de fichiers de l'arborescence des fichiers SPL2 « zippés »	42
Table 16:	Taille et nombre de fichiers de l'arborescence des fichiers PNG»	43
Table 17:	Résumé des volumes occupés par les bases de données locales (Go).....	44
Table 18:	Exemple de résultat de la commande RCL_list	45
Table 19:	Exemple de résultat de la commande RCL_list_block_WF.....	48
Table 20:	Exemple de résultat de la commande RCL_give_CLUORB_period	50
Table 21:	Exemple de résultat de la commande RCL_give_spin_dir	54
Table 22:	Exemple de résultat de la commande RCL_system_info	64
Table 23:	Exemple de résultat de la commande RCL_dir_size_tree	65
Table 24:	Exemple de résultat de la commande RCL_dir_size_tree Mo.....	65
Table 25:	Exemple de résultat de la commande RCL_dir_size_pretty_tree	66
Table 26:	Exemples de résultats de la commande RCL_dir_properties_tree	67
Table 27:	Exemple de résultat de la commande RCL_dir_properties_pretty_tree	68
Table 28:	Exemple de résultat de la commande RCL_list	70
Table 29:	Contenu du script «PR_make_staff_spectro_from_VTL2 »	72
Table 30:	Contenu du script «PR_make_staff_spectro_from_VTL1 »	75

Table 31:	Contenu du script «PR_visu_staff_VTL1 »	77
Table 32:	Contenu du script «PR_visu_staff_VTL2 »	78
Table 33:	Contenu du script « PR_compute_staff_polarisation_from_VTL1 »	82
Table 34:	Exemple de script de traitement de masse pour l'étude de la polarisation	83
Table 35:	Contenu du script « PR_compute_staff_polarisation_from_VTL2 »	85
Table 36:	Script permettant de calculer et visualiser un spectrogramme de FGM	86
Table 37:	Exemple de fichier RCL_config.bash.....	90
Table 38:	Options de compilation du fichier Makefile.....	92
Table 39 :	Options « make clean » du fichier Makefile.....	92
Table 40:	Options « make mrproper » du fichier Makefile	93
Table 41:	Compilation desIDL.pro dans le fichier Make_IDL_sav.bash	93
Table 42:	Arborescence des répertoires créés par le test de production sur un mois.....	95
Table 43:	Arborescence des répertoires du projet RCL.....	97
Table 44:	Fichiers à la racine du projet RCL.....	97
Table 45:	Liste des commandes RCL utilisant les exécutable du répertoire bin.....	100
Table 46:	Liste des commandes RCL utilisant des SAV d'IDL du répertoire bin	101
Table 47:	Commandes RCL utilisant d'autres commandes RCL.....	103
Table 48:	Bibliothèques utilisées par chaque programme exécutable F90	109
Table 49 :	Récapitulation des bibliothèque F90 utilisées	110
Table 50 :	Liste des sousroutines de traitement du signal de la bibliothèque lib_déconvo.f90	110
Table 51 :	Liste des sousroutines de la bibliothèque lib_gainant.f90	110
Table 52 :	Liste des sousroutines de la bibliothèque lib_rw_rff.f90	111
Table 53 :	Liste des sousroutines de la bibliothèque lib_rw_cef.....	112
Table 54 :	Liste des sousroutines de la bibliothèque lib_time.f90	112
Table 55 :	Liste des sousroutines de la bibliothèque lib_CLUORB.f90	113
Table 56 :	Liste des sousroutines de la bibliothèque lib_CLU_tools.f90	113
Table 57 :	Liste des sousroutines de la bibliothèque lib_utility.f90	114
Table 58 :	Liste des sousroutines de la bibliothèque Rocolib_2p0.f90.....	115
Table 59:	Procédures IDL de lecture de fichiers.....	116
Table 60 :	Procédures IDL de visualisation de fichiers de données	117
Table 61:	Exemple de fabrication d'un .sav d'IDL	118
Table 62:	Liste des procédures IDL nécessaire à la fabrication des .SAV.....	119
Table 63 :	Procédures et fonctions disponibles dans la bibliothèque roploplib.pro.....	121
Table 64 :	STAFF status word.....	143
Table 65 :	Signification de chacun des 14 caractères du statut des VTL1 ou des DWF	144
Table 66 :	Erreur maximum du à la compression.....	145
Table 67:	Exemple de WFL1.rff (3 pages).....	175
Table 68:	Exemple de VTL1.rff (3 pages).....	178
Table 69 :	Exemple de VTL2.rff (3 pages).....	181
Table 70:	Exemple de SPL2.rff (3 pages).....	184
Table 71:	Liste fonctionnelle des commandes RCL (3 pages)	205
Table 72:	Liste alphabétique des commandes RCL avec arguments (3 pages).....	208
Table 73 :	Introduction aux Roproc	210
Table 74:	Liste thématique des Roproc (6 pages).....	216
Table 75:	Exemples of arguments of Roproc commands (2 pages).....	218

12. BIBLIOGRAPHIE

- [1] Patrick Robert, “Les procédures Roproc pour le traitement des données de CLUSTER /STAFF-SC, FGM et ORBITE”, CETP, V 2.0, Septembre 2003.
ftp://ftp.lpp.polytechnique.fr/robert/keep/ROPROC/Roproc_V2p0.pdf
- [2] C. Perry, T. Eriksson, P. Escoubet, S. Esson, H. Laakso, S. McCaffrey, T. Sanderson, H. Bowen, “**The ESA Cluster Active Archive**”, in *Proceedings of the Cluster and Double Star Symposium 5th Anniversary of Cluster in Space, ESTEC*, Noordwijk, 2005.
<http://caa.estec.esa.int/caa/home.xml>
- [3] A. Allen, S.J.Schwartz, C. Harvey, C. Perry, C. Huc, P. Robert, “**Cluster Exchange Format - Data File Syntax**”, CSDS Archive Task Group, October 19, 2009.
<http://caa.estec.esa.int/documents/DS-QMW-TN-0010.pdf>
- [4] Patrick ROBERT, “**The Roproc File Format, A dedicated file format for vectorial data processing**” Version 2.2, February 2011, First Version 1.0, November 2004, CNRS/LPP
ftp://ftp.lpp.polytechnique.fr/robert/keep/ROPROC/The_Roproc_File_Format_20110216.pdf
- [5] Patrick ROBERT, “**Roproc Short Documentation**”, Version 4.3 CNRS/CETP-LPP, 2000-2010, Last update January 11, 2011
ftp://ftp.lpp.polytechnique.fr/robert/keep/ROPROC/Short_doc_Roproc_V4p3_c.pdf
- [6] Nicole Cornilleau Wehrlin : Chauveau, P.: Louis, S.: Meyer, A.: Nappa, J. M.: Perraut S.: Rezeau, L.: Robert, P.: Roux, A. and C. De Villerdary: “**The cluster spatiotemporal analysis of field fluctuations (STAFF) experiment**”. *Space Science Review* 79: 107- 136, 1997.
- [7] N. Cornilleau-Wehrin, C. Burlaud and the STAFF team, “**User Guide to the STAFF measurements in the Cluster Active Archive (CAA)**”, ESA CAA-EST-UG-002, April 26, 2011.
http://caa.estec.esa.int/documents/UG/CAA_EST_UG_STA_v30.pdf
- [8] Patrick Robert, “**ROCOTLIB: a Coordinate Transformation Library for SolarTerrestrial studies**”, version 1.8, update of CETP Internal report n° RI-CETP/01/2003, Novembre 2003.
ftp://ftp.lpp.polytechnique.fr/robert/keep/ROCOTLIB/ROCOTLIB_v1.8u_TOC.pdf
- [9] xxx, “Description de la chaîne de décommutation des données STAFF-SC”, version ..., année...
- [10] C.C. Harvey, A.J. Allen, F. D'ariot, C. Huc, M. Nonon-Latapie, C.H. Perry, S.J. Schwartz, T. Eriksson and S. McCaffrey, “**Cluster Metadata Dictionary**”, CAA Metadata Working Group, March 5, 2008.
<http://caa.estec.esa.int/documents/DataDic.pdf>

- [11] Patrick ROBERT, C. Burlaud & STAFF Team, “**STAFF/SC cross-calibration activities**”, 10th CAA Cross-Calibration meeting, Paris, 2-4 November 2009
<ftp://ftp.rssd.esa.int/pub/Cluster/MoM/CAA/CrossCalibration/CrossCal10/CrossCal10-Annex21-STAFF-SC.ppt>
- [12] Patrick ROBERT, “**STAFF CAA products & Cross-Calibration activities**”, 13th CAA Cross-Calibration meeting, Uppsala, 13-15 Apr 2011
- [13] P. Robert et al., “**CLUSTER STAFF search coils magnetometer calibration - Comparisons with FGM**”, in Special Issue: Calibration methods and results of the in-situ experiments on Cluster and Double Star, Research Article, gi-2013-15
<http://www.geosci-instrum-method-data-syst.net/3/153/2014/>
- [14] P. Robert., A. Roux, C.C. Harvey, M.W. Dunlop, P.W. Daly, and K.H. Glassmeier, “**Tetrahedron geometric Factors**”, Analysis Method for Multi-Spacecraft Data, Eds. G. Paschmann and P.W. Daly, International Space Science Institute, Bern, Switzerland, pp. 323-348, 1998. (ESA Publication Division, SR-001, July 1998).
- [15] Chanteur, G. and Mottez, F., “**Geometrical tools for Cluster data analysis**”, in Proc. International Conf. “Spatio-Temporal Analysis for Resolving plasma Turbulence (START)”, Aussois, 31 Jan.–5 Feb. 1993 , ESA WPP-047, pp. 341–344, European Space Agency, Paris, France, 1993.
- [16] Patrick Robert and C. de Villedary , “**Transformation of a STAFF waveform into a Magnetic Field Aligned coordinate system**”, Rapport interne CNRS-UVSQ/CETP n° RICETP/6/2000, Octobre 2000.
- [17] ESA/ESOC, “**Data Delivery Interface Document (DDID)**”, Directorate of Technical and Operational Support, European Space Operations Centre, CLUSTER Data Disposition System, Issue 3, Document Reference: CL-ESC-ID-2001, 19 May 2000.
- [18] Patrick Robert, “**A, easy method to compute plane waves polarisation from a three components waveform**”, Document de travail CRPE, 1980. Révision et application à CLUSTER, Aout 2008.
- [19] Patrick Robert, C. Burlaud and M. Maksimovic, “**Calibration Report of the STAFF Measurements in the Cluster Active Archive (CAA)**”, Version 3.0, Updated by N. Cornilleau-Wehrin, P. Robert and R. Piberne, doc. CAA/ESA Num. CAA-STA-CR-002, Issue: 3.0, 2012-05-16.
http://caa.estec.esa.int/documents/CR/CAA_EST_CR_STA_v30.pdf
- [20] Robert, P.: « **Intensité et polarisation des ondes UBF détectées à bord de GEOS-1, Méthode d’analyse numérique du signal et production en routine de sommaires expérimentateurs, Problèmes rencontrés et solutions pratiques** », Note Technique CRPE/ETE/71, May, available at:
ftp://ftp.lpp.polytechnique.fr/robert/keep/Biblio_et_CV/Working_documents/1979_Robert_NTCRPE71_Intensite_et_Polarisation_des_ondes_UBF.pdf

- NOTES -

— NOTES —