

# Vimpack

- Get last version of file
- Automatic update of current branch (« local »)
- Code browser
- Fast searching

# File editing

\$ vimpack cnt4.F90

```
1 SUBROUTINE CNT4
2
3 !**** *CNT4* - Controls integration job at level 4
4
5 ! Purpose.
6 ! -----
7 ! CONTROLS THE INTEGRATION
8
9 !** Interface.
10 ! -----
11 ! *CALL* *CNT4
12
13 ! Explicit arguments :
14 ! -----
15 ! None
16
17 ! Implicit arguments :
18 ! -----
19 ! None
20
21 ! Method.
22 ! -----
23 ! See documentation
24
25 ! Externals.
```

".vimpack/src=/arpifs/control/cnt4.F90" 1301 lines --0%-- 1,1 Top

# Update, automatic copy to local

```
369 ! -----
370
371 !*      2.      Prepare occurrences of I/O events.
372 ! -----
373
374 CALL MONIO(IPOSTS,IPISPS,IHISTS,ISFXHISTS,IGDITS,ISDITS,IDHFGTS,IDHFZTS,&
375 & IDHFDTS,IDHPTS,ICFUTS,IXFUTS,IRESTS,IMASSCONS,IRAZTS)
376 CALL MONVAR(IREFTS,IANATS,IGRATS)
377 IF (LELAM) THEN
378     CALL MOEVAR(ILSGTS)
379 ENDIF
380
381 X = X + 1
382
383 IF (NPRINTLEV >= 1) THEN
384     WRITE(NULOUT, '(' POST-PROCESSING EVENTS, IPOSTS ')')
385     WRITE(NULOUT, '(40I2)') IPOSTS
386     WRITE(NULOUT, '(' ISP (Animation !) EVENTS, IPISPS ')')
387     WRITE(NULOUT, '(40I2)') IPISPS
388     WRITE(NULOUT, '(' HISTORY WRITE-UP, IHISTS ')')
389     WRITE(NULOUT, '(40I2)') IHISTS
390     WRITE(NULOUT, '(' HISTORY SURFACE WRITE-UP, ISFXHISTS ')')
391     WRITE(NULOUT, '(40I2)') ISFXHISTS
392     WRITE(NULOUT, '(' MASS CONSERVATION FIXUP, IMASSCONS ')')
393     WRITE(NULOUT, '(40I2)') IMASSCONS

```

"src/local/arpifs/control/cnt4.F90" 1303L, 39241C 381.9 28%

# Browsing

```
369 ! -----
370
371 !* 2. Prepare occurrences of I/O events.
372 ! -----
373
374 CALL MONIO(IPOSTS,IPISPS,IHISTS,ISFXHISTS,IGDITS,ISDITS,IDHFGTS,IDHFZTS,&
375 & IDHFDTS,IDHPTS,ICFUTS,IXFUTS,IRESTS,IMASSCONS,IRAZTS)
376 CALL MONVAR(IREFTS,IANATS,IGRATS)
377 IF (LELAM) THEN
378 CALL MOEVAR(ILSGTS)
379 ENDIF
380
381 X = X + 1
382
383 IF (NPRINTLEV >= 1) THEN
384 WRITE(NULOUT,(' POST-PROCESSING EVENTS, IPOSTS '))
385 WRITE(NULOUT,(' (40I2)')IPOSTS
386 WRITE(NULOUT,(' ISP (Animation !) EVENTS, IPISPS '))
387 WRITE(NULOUT,(' (40I2)')IPISPS
388 WRITE(NULOUT,(' HISTORY WRITE-UP, IHISTS '))
389 WRITE(NULOUT,(' (40I2)')IHISTS
390 WRITE(NULOUT,(' HISTORY SURFACE WRITE-UP, ISFXHISTS '))
391 WRITE(NULOUT,(' (40I2)')ISFXHIST
392 WRITE(NULOUT,(' MASS CONSERV
393 WRITE(NULOUT,(' (40I2)')IMASSCON
"src/local/arpifs/control/cnt4.F90" 1
```

CALL MONIO  
USE YOMCT0  
TYPE (IOFLDDDESC)

```
1 SUBROUTINE MONIO(KPOSTS,KPISPS,KHISTS,KSFXHISTS,KGDITS,KSDITS,KDHFGTS,&
2 & KDHFZTS,KDHFDTS,KDHPTS,KCFUTS,KXFUTS,KRESTS,KMASSCONS,KRAZTS)
3
4 !**** *MONIO* - Management of the I/O events
5
6 ! Purpose.
7 ! -----
8 ! Set up the I/O events control arrays
9
10 !** Interface.
11 ! -----
12 ! *CALL* MONIO(...)
13
14 ! Explicit arguments :
15 ! -----
16 ! KPOSTS : ARRAY CONTAINING POST-PROCESSING TIME STEPS
17 ! KPISPS : ARRAY CONTAINING ISP (Animation !) TIME STEPS
18 ! KHISTS : ARRAY CONTAINING TRAJECTORY TIME STEPS
19 ! KSFXHISTS: ARRAY CONTAINING TRAJECTORY TIME STEPS FOR SURFACE
20 ! KGDITS : GRID POINT DIAGNOSTICS TIME STEPS
21 ! KSDITS : SPECTRAL DIAGNOSTICS TIME STEPS
22 ! KDHFGTS : WRITE OUT TIME STEPS FOR GLOBAL MEANS DDH
23 ! KDHFZTS : WRITE OUT TIME STEPS FOR ZONAL MEANS DDH
24 ! KDHFDTS : WRITE OUT TIME STEPS FOR LIMITED AREAS DDH
25 ! KDHPTS : PAS DE TEMPS IMPRESSIONS DIAGNOSTICS DDH
```

"vimpack/src=arpifs/control/monio.F90" 225L, 6483C

1.1

Top

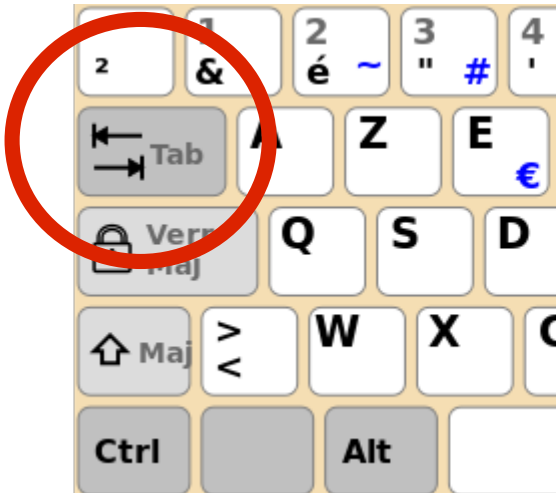
# Searching

```
773 & .OR. NSIM4D == NSIM4DL) )
774
775 !* 3.9.1 Variational analysis on
776
777 IF(NCONF/100 == 1.OR.LLTLEVOL) THEN
778   CLCONF(1:9)='A00000000'
779   RESTART=CLCONF(1:1)
780   LTWANA=.TRUE.
781   LLFDBOP=LFDBOP
782   LFDBOP=.FALSE.
783   IOUTTYPE=NOUTTYPE
784   NOUTTYPE=1
785   CALL STEP0(CLCONF)
786   IF (LLTLEVOL) THEN
787     CALL PPFLUSH
788     CALL PPCLOSE
789   ENDIF
790   LFDBOP=LLFDBOP
791   NOUTTYPE=IOUTTYPE
792   LTWANA=.FALSE.
793 ENDIF
794
795 !* 3.9.2 Post-processing
796
797 IF(LLWRTRA) THEN
```

```
1
2 aladin/coupling/ecoupl1.F90
3
4 237 | ENDDO ! JGPBLKS
5 238 |
6 239 | ! "time stepping" for coupled grid-point GFL fields (must go in STEP0 and STEPOTL).
7 240 | !$OMP do
8 241 | DO JFL=1,YGFL%NUMFLDS
9
10 aladin/sinvect/ewrtsv.F90
11
12 66 | #include "cain.intfb.h"
13 67 | #include "scaas.intfb.h"
14 68 | #include "step0.intfb.h"
15 69 |
16 70 | ! -----
17 ...
18 82 | LTWGRA=.FALSE.
19 83 | NSIM4D=KEIG
20 84 | call STEP0("A00000000")
21 85 | LTWLCZ=.FALSE.
22 86 |
23
24 aladin/utility/euvcopy.F90
25
26 34 | ! Modifications.
27 35 | ! -----
28 36 | ! Original : 98-03-30 From STEP0
29 37 | ! M.Hamrud 01-Oct-2003 CY28 Cleaning
30 38 | ! A.Bogatchev 15-04-2013 NFLEVL => YOMDIMV
31
32 aladin/var/ecosjr.F90
33
34 88 | ! -----
35 89 |
36 90 | #include "step0.intfb.h"
37 91 | #include "stepoad.intfb.h"
38 92 | #include "stepotl.intfb.h"
39 ...
40 126 | ! Mask Operator
41 127 | IF(LOCNORM) THEN
42 128 | CALL STEP0 ('OK0000000')
43 129 | CALL STEPOTL('OGB0LOAA0')
44 130 | ENDIF
45
46 aladin/var/ewreini.F90
47
48 13 | ! PDX - initial perturbation
49 14 |
50 15 | ! Externals - SIM4D, STEP0
51 16 | ! -----
52 17 |
53 ...
'step0' was found in 48 files
```

# Browsing

```
1
2 aladin/coupling/ecoupl1.F90
3
4 237 | ENDDO ! JGPBLKS
5 238 |
6 239 | ! "time stepping" for coupled grid-point GFL fields (must go in STEP0 and STEP0TL).
7 240 | !$OMP do
8 241 | DO JFL=1,YGFL%NUMFLDS
9
10 aladin/sinvect/ewrtsv.F90
11
12 66 | #include "cain.intfb.h"
13 67 | #include "scaas.intfb.h"
14 68 | #include "stepo.intfb.h"
15 69 |
16 70 | ! -----
17
18 82 | LTWGRA=.FALSE.
19 83 | NSIM4D=KEIG
20 84 | call STEP0("A00000000")
21 85 | LTWLCZ=.FALSE.
22 86 |
23
24 aladin/utility/euvcopy.F90
25
26 34 | ! Modifications.
27 35 | ! -----
28 36 | ! Original : 98-03-30 From STEP0
29 37 | ! M.Hamrud 01-Oct-2003 CY28 Cleaning
30 38 | ! A.Bogatchev 15-04-2013 NFLEVL => YOMDIMV
31
32 aladin/var/ecosjr.F90
33
34 88 | ! -----
35 89 |
36 90 | #include "stepo.intfb.h"
37 91 | #include "stepoad.intfb.h"
38 92 | #include "stepotl.intfb.h"
39 ...
40 126 | ! Mask Operator
41 127 | IF(LOCNORM) THEN
42 128 | CALL STEP0('0K0000000')
43 129 | CALL STEP0TL('0GB0L0AA0')
44 130 | ENDIF
45
46 aladin/var/ewreini.F90
47
48 13 | ! PDX - initial perturbation
49 14 |
50 15 | ! Externals - SIM4D, STEP0
51 16 | ! -----
52 17 |
53 ...
'step0' was found in 48 files
```



# Difference

<pre>+ 1 +--126 lines: #####- 127  ALLOCATE(ZFIELDOUT(SIZE(XLAT),SIZE(ZFIELDIN,2))) 128  ALLOCATE(ZFIELDOUT(SIZE(XLAT),1)) 129   130 !Impossible to interpolate lake profiles, only the lake surface temperature! 131 !But in uniform case and 1 point case 132 IF(GUNIF .OR. SIZE(XLAT).EQ.1) THEN ----- 133 CALL HOR_INTERPOL(ILUOUT,ZFIELDIN,ZFIELDOUT) 134 ELSE IF(HSURF(1:2)=='ZS' .OR. HSURF(1:2)=='TS') THEN 135 WRITE(ILUOUT,*) "WARNING! Impossible to interpolate lake profiles in horizontal!" 136 WRITE(ILUOUT,*) "So, interoplate only surface temperature and start from lakes mix ----- 137 CALL HOR_INTERPOL(ILUOUT,ZFIELDIN,ZFIELDOUT) 138 END IF 139   140 !*      5.      Return to historical variable 141   142 SELECT CASE (HSURF) + 143 +- 61 lines: CASE('ZS    ') ----- ~ ~ ~ ~ ~ ~ ~ ~ ~ ~</pre>	<pre>+ 1 --126 lines: #####- 127  ALLOCATE(ZFIELDOUT(SIZE(XLAT),SIZE(ZFIELDIN,2))) 128  ALLOCATE(ZFIELDOUT(SIZE(XLAT),1)) 129   130 !Impossible to interpolate lake profiles, only the lake surface temperature! 131 !But in uniform case and 1 point case 132 IF(GUNIF .OR. SIZE(XLAT).EQ.1) THEN ----- 133 WRITE (ILUOUT, *) HSURF 134 CALL HOR_INTERPOL(ILUOUT,ZFIELDIN,ZFIELDOUT) 135 ELSE IF(HSURF(1:2)=='ZS' .OR. HSURF(1:2)=='TS') THEN 136 WRITE(ILUOUT,*) "WARNING! Impossible to interpolate lake profiles in horizontal!" 137 WRITE(ILUOUT,*) "So, interoplate only surface temperature and start from lakes mi ----- 138 WRITE (ILUOUT, *) HSURF 139 CALL HOR_INTERPOL(ILUOUT,ZFIELDIN,ZFIELDOUT) 140 END IF 141   142 !*      5.      Return to historical variable 143   144 SELECT CASE (HSURF) + 145 +- 61 lines: CASE('ZS    ') ----- ~ ~ ~ ~ ~ ~ ~ ~ ~ ~</pre>
--	--

```
$ vimpack -d prep_hor_flake_field.F90
or
:D in vim command line
```

# Compilateur messages

```
141 CALL GATH_GRID (KFGATHG=KFIELDS,KTO=ITO,PGP=PBUF,PGPG=ZGPG)
142 ENDIF
143
144 DO JFLD = 1, INFL
145   IFLDG = IFLDOFF(MYPROC) + JFLD
146   IF (YDFLDSC (IFLDG)%LUNDF) &
147     & CALL WRGP2FA_REMOVE_UNDEF (ZGPG (:,JFLD), YDFLDSC(IFLDG))
148 ENDDO
149
150 CALL FPGPNORMX (ZGPG, YLCPDSC%ZAVE, YLCPDSC%ZMIN, YLCPDSC%ZMAX)
151
152 <error:
153 A logical data type is required in this context. [LDOCQ]
154 This is not a field name that is defined in the encompassing structure. [LDOCQ]
155 }>IF (YDFACTX%YI0OPTS%LDOCQ) THEN
156
157   IDIMGVAL=MAXVAL (YDFLDSC%NSIZPK)
158   ALLOCATE (ZGVALCO (IDIMGVAL, INFL))
159
160   IFLDG1=IFLDOFF(MYPROC)+1
161   IFLDG2=IFLDOFF(MYPROC)+INFL
162
163   CALL COMPACTFLD (YDFACTX, YDFLDSC (IFLDG1:IFLDG2), ZGPG, YLCPDSC, ZGVALCO)
164
165   CALL WRGATHFLNM (1_JPIM, INFL, IDIMGVAL, INFD, IFLDOFF, ZGVALCO, YLCPDSC, &
166     & LDNORM=.TRUE., YDFACTX=YDFACTX, KTAG=KTAG, &
167     & CDNORMTITLE='GPNORMS OF FIELDS TO BE WRITTEN OUT ON FILE :')
168
169   DEALLOCATE (ZGVALCO)
```

152,1

93%

\$ vimpack -w wrgp2fa.F90

or

:WI (vim command)



# Online help

```
Gmckpack and Vim                                *vimpack*

1. Description                                |vimpack-description|
2. Usage                                      |vimpack-usage|
3. Principles                                |vimpack-principles|
4. Commands and mappings                    |vimpack-commands|
5. History                                  |vimpack-history|

=====
1. Description                                *vimpack-description*

A pack editor based on vim; a vim compiled with the embedded Perl interpreter
has to be available in your PATH.

=====
2. Usage                                      *vimpack-usage*

You must first cd to a valid pack. An index must have been build using the
`vimpack -i' command. Then just type `vimpack filename.F90'; for instance >

vimpack cnt0.F90

<

To edit in diff mode (assuming cnt0.F90 has been modified in an intermediate
or local pack) >
vimpack.txt [Help][R0]                        5.53                32
1
[No Name]                                     0.0-1                All
"vimpack.txt" [readonly] 149L, 4507C
```

:help vimpack (vim command)