

MISO : Modifying the IFS Source for OOPS

Olivier Marsden

OOPS Presentation February 2016

- SPAMming the IFS
 - Script for Passing Arguments Modification
 - *DONE* geometry & fields
 - *TO DO* model configuration
- New trajectory code for OOPS

Passing Data by Argument

OOPS requires data and parameters related to a task to be objects it knows of and can manipulate.

Until recently, the IFS stored most of its mutable data in modules, and routines accessed this data implicitly via `USE MODULE, ONLY : DATA`

A necessary but not sufficient step towards functional OOPS has been to modify all IFS routines to work on data passed as arguments.

Passing Data by Argument

OOPS requires data and parameters related to a task to be objects it knows of and can manipulate.

Until recently, the IFS stored most of its mutable data in modules, and routines accessed this data implicitly via `USE MODULE, ONLY : DATA`

A necessary but not sufficient step towards functional OOPS has been to modify all IFS routines to work on data passed as arguments.

Previous analysis (OOPS framework, T. Wilhelmsson *et al*) determined that there would be three families of data to be dealt with :

- geometry-related data
- flow field data
- model configuration data

To this end, a python script was written which

- analyses the whole IFS source tree
- generates a map of which modules are used in which routines
- generates a call tree of where individual routines are called
- uses this information to remove offending module usage, and replace it with appropriate geometry, field or model argument
- has been used at ECMWF to modify a large fraction of the files in the IFS/Arpège source tree, and LAM-specific code on the Météo-France side

Example

With T. Wilhelmsson's past encapsulation work, many routines look like this :

```
SUBROUTINE DOES__NOTHING
! great routine with thoughtful comments

USE PARKIND1 , ONLY : JPIM
USE YOMDIM , ONLY : YRDIM
USE YOMDIMV , ONLY : YRDIMV
IMPLICIT NONE

INTEGER(KIND=JPIM) :: USELESS_INT
ASSOCIATE(NICE_NAME1=>YRDIM%NGPBLKS,
NICE_NAME2=>YRDIMV%NFLEVG)

IF (NICE_NAME1== -1 ) THEN
CALL ANOTHER_ROUTINE
END IF

...
```

Example

With T. Wilhelmsson's past encapsulation work, many routines look like this :
and SPAM updates it to this:

```
SUBROUTINE DOES_NOTHING
```

```
! great routine with thoughtful comments
```

```
USE PARKIND1 , ONLY : JPIM
```

```
USE YOMDIM , ONLY : YRDIM
```

```
USE YOMDIMV , ONLY : YRDIMV
```

```
IMPLICIT NONE
```

```
INTEGER(KIND=JPIM) :: USELESS_INT
```

```
ASSOCIATE(NICE_NAME1=>YRDIM%NGPBLKS,  
NICE_NAME2=>YRDIMV%NFLEVG)
```

```
IF (NICE_NAME1== -1 ) THEN
```

```
CALL ANOTHER_ROUTINE
```

```
END IF
```

```
...
```

```
SUBROUTINE DOES_NOTHING_BUT_BETTER(YDGEOM)
```

```
! great routine with thoughtful comments
```

```
USE GEOMETRY_MOD , ONLY : GEOMETRY
```

```
USE PARKIND1 , ONLY : JPIM
```

```
IMPLICIT NONE
```

```
TYPE(GEOMETRY),INTENT(IN) :: YDGEOM
```

```
INTEGER(KIND=JPIM) :: USELESS_INT
```

```
ASSOCIATE(NICE_NAME1=>YDGEOM%YRDIM%NGPBLKS,  
NICE_NAME2=>YDGEOM%YRDIMV%NFLEVG )
```

```
IF (NICE_NAME1== -1) THEN
```

```
CALL ANOTHER_ROUTINE(NICE_NAME2)
```

```
END IF
```

```
...
```

Current state and outlook

Done so far:

- FIELDS (GMV, GFL, SURF) done for CY42
- GEOMETRY (most of it) was done at CY42R2
- GEOMETRY (rest of it) now done for CY42R3

Current state and outlook

Done so far:

- FIELDS (GMV, GFL, SURF) done for CY42
- GEOMETRY (most of it) was done at CY42R2
- GEOMETRY (rest of it) now done for CY42R3

Stay tuned, more to come:

- MODEL will be done at some point (CY43RX)

Current state and outlook

Done so far:

- FIELDS (GMV, GFL, SURF) done for CY42
- GEOMETRY (most of it) was done at CY42R2
- GEOMETRY (rest of it) now done for CY42R3

Stay tuned, more to come:

- MODEL will be done at some point (CY43RX)
- MODEL comprises of many dozens of further modules, making some form of hierarchical treatment necessary

Current state and outlook

Done so far:

- FIELDS (GMV, GFL, SURF) done for CY42
- GEOMETRY (most of it) was done at CY42R2
- GEOMETRY (rest of it) now done for CY42R3

Stay tuned, more to come:

- MODEL will be done at some point (CY43RX)
- MODEL comprises of many dozens of further modules, making some form of hierarchical treatment necessary
- current (**non-definitive**) plan is to group modules into following groups :

Current state and outlook

Done so far:

- FIELDS (GMV, GFL, SURF) done for CY42
- GEOMETRY (most of it) was done at CY42R2
- GEOMETRY (rest of it) now done for CY42R3

Stay tuned, more to come:

- MODEL will be done at some point (CY43RX)
- MODEL comprises of many dozens of further modules, making some form of hierarchical treatment necessary
- current (**non-definitive**) plan is to group modules into following groups :

Current state and outlook

Done so far:

- FIELDS (GMV, GFL, SURF) done for CY42
- GEOMETRY (most of it) was done at CY42R2
- GEOMETRY (rest of it) now done for CY42R3

Stay tuned, more to come:

- MODEL will be done at some point (CY43RX)
- MODEL comprises of many dozens of further modules, making some form of hierarchical treatment necessary
- current (**non-definitive**) plan is to group modules into following groups :

General model config
Ocean / ice coupling
Dynamics

Current state and outlook

Done so far:

- FIELDS (GMV, GFL, SURF) done for CY42
- GEOMETRY (most of it) was done at CY42R2
- GEOMETRY (rest of it) now done for CY42R3

Stay tuned, more to come:

- MODEL will be done at some point (CY43RX)
- MODEL comprises of many dozens of further modules, making some form of hierarchical treatment necessary
- current (**non-definitive**) plan is to group modules into following groups :

General model config
Ocean / ice coupling
Dynamics

Physics
Diagnostics

Current state and outlook

Done so far:

- FIELDS (GMV, GFL, SURF) done for CY42
- GEOMETRY (most of it) was done at CY42R2
- GEOMETRY (rest of it) now done for CY42R3

Stay tuned, more to come:

- MODEL will be done at some point (CY43RX)
- MODEL comprises of many dozens of further modules, making some form of hierarchical treatment necessary
- current (**non-definitive**) plan is to group modules into following groups :

General model config
Ocean / ice coupling
Dynamics

Physics
Diagnostics

- currently working on new trajectory handling (new code path, so minimal interference with current IFS code) for OOPS, being tested at the moment

Clean-ups / cleans-up to come

Ordered by increasing importance:

- **ASSOCIATE** statements currently contain a lot of unused associations (SPAM)

Ordered by increasing importance:

- **ASSOCIATE** statements currently contain a lot of unused associations (SPAM)
- arguments added by SPAM mostly have intent INOUT, whereas in many places they could (should) be IN

Ordered by increasing importance:

- **ASSOCIATE** statements currently contain a lot of unused associations (SPAM)
- arguments added by SPAM mostly have intent INOUT, whereas in many places they could (should) be IN
- many routines now have more than one argument providing access to the same data this will require thought as to how best to correct
Examples :

Ordered by increasing importance:

- **ASSOCIATE** statements currently contain a lot of unused associations (SPAM)
- arguments added by SPAM mostly have intent INOUT, whereas in many places they could (should) be IN
- many routines now have more than one argument providing access to the same data this will require thought as to how best to correct

Examples :

- **SUBROUTINE DOWORK(..., YDFIELDS, PGMV, ...)**

Ordered by increasing importance:

- **ASSOCIATE** statements currently contain a lot of unused associations (SPAM)
- arguments added by SPAM mostly have intent INOUT, whereas in many places they could (should) be IN
- many routines now have more than one argument providing access to the same data this will require thought as to how best to correct

Examples :

- **SUBROUTINE DOWORK(..., YDFIELDS, PGMV, ...)**
- **SUBROUTINE DOWORK(YDGEOMETRY,YDFIELDS, ...)**

Example illustrating difficulties

Subroutine ec_phys_drv.F90

```
SUBROUTINE EC_PHYS_DRV(YDGEOMETRY, YDGMV, YDSURF, CDCONF, PB2, PPBUP9, PPBVP9, PPBTP9, PPBGFLP9, PEXTRADYN, PSD_PF, &  
  & PGMV, PGMVS, PGMVT1, PGMVT1S, &  
  & PGFLT1, PGFLPT, PGFL, PGFLSLP, &  
  & PSP_SB, PSP_SG, PSP_SL, PSP_RR, &  
  & PSP_EP, PSP_X2, &  
  & PSD_VF, PSD_VN, PSD_VD, &  
  & PSD_WS, PSD_WW, &  
  & PSD_XA, PSD_X2, &  
  & PSP_OM, PSD_V2, PSD_V3, &! KPP  
  & PSMOS_OBS_BUF, PSMOS_TB_BUF, PSMOS_ANGLE, PSMOS_FARAD, PSMOS_GEOMET, &  
  & KSMOS_BUF, &  
  & PGBRAD_OBS_BUF, PGBRAD_RR_BUF, KGBRAD_BUF, &  
  & PRAINGG_OBS_BUF, PRAINGG_RR_BUF, KRAINGG_BUF, PTRAJEC)
```