Science and Technology Facilities Council

Investigating Interoperability

Rupert Ford, Andy Porter, Sergi Siso, STFC Hartree Centre
Iva Kavcic, Chris Maynard, Andrew Coughtrie, UK Met Office
Joerg Henrichs, Australian Bureau of Meteorology

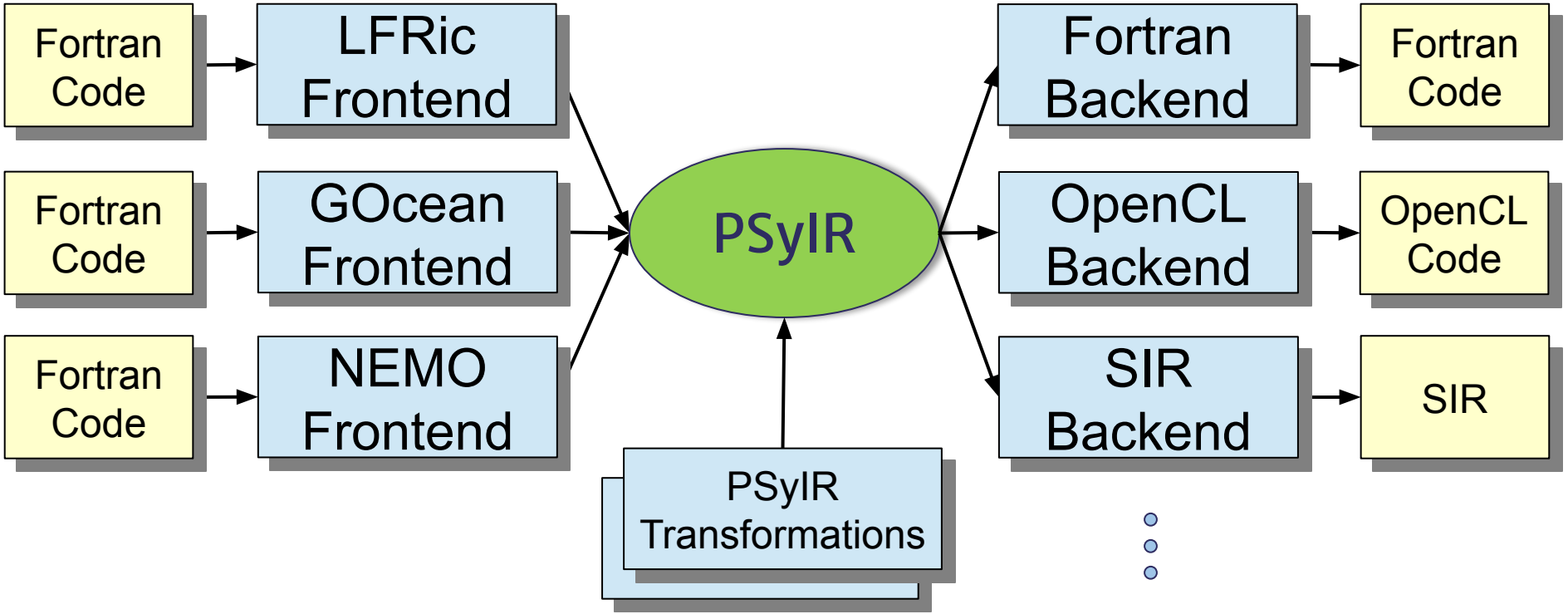
Hartree Centre
Science & Technology Facilities Council

Met Office

Australian Government
Bureau of Meteorology

ESIWACE2 training course on Domain-specific Languages in Weather and Climate, 23rd-27th November 2020

PSyclone front- and back-ends



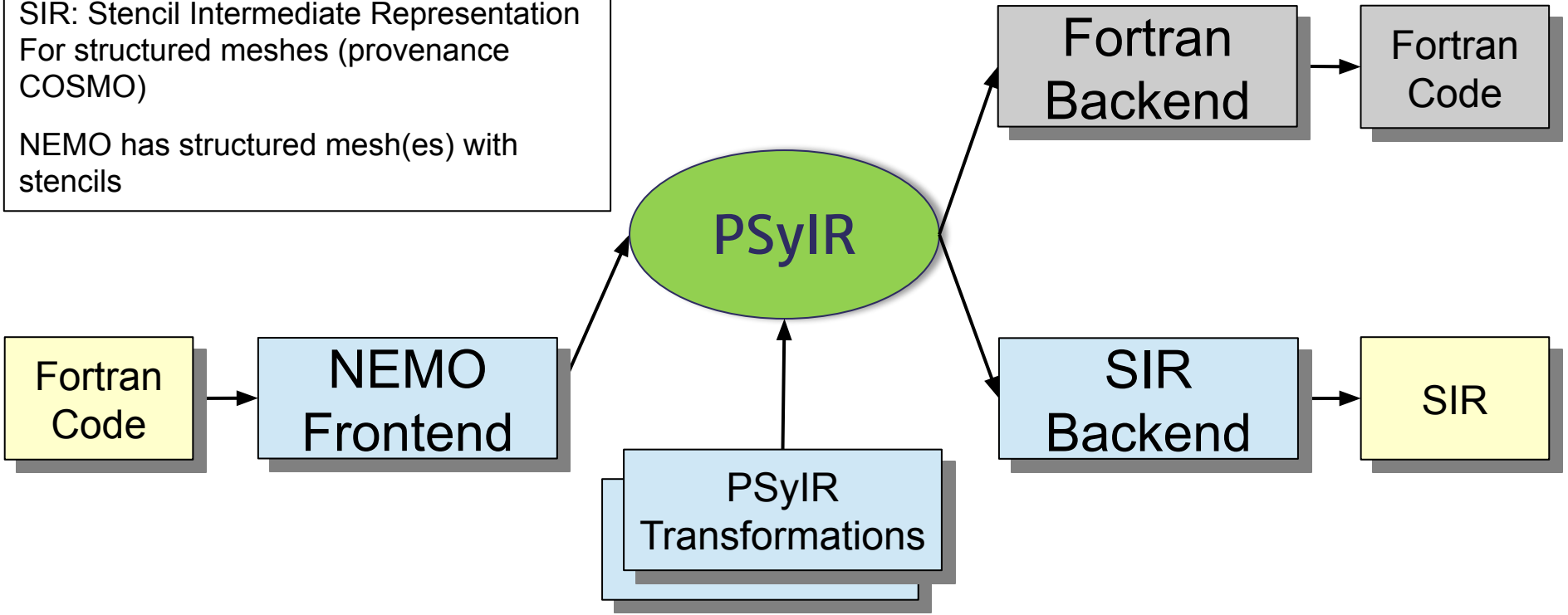
ESiWACE2 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823988

esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE

SIR back-end for NEMO front-end

SIR: Stencil Intermediate Representation
For structured meshes (provenance COSMO)

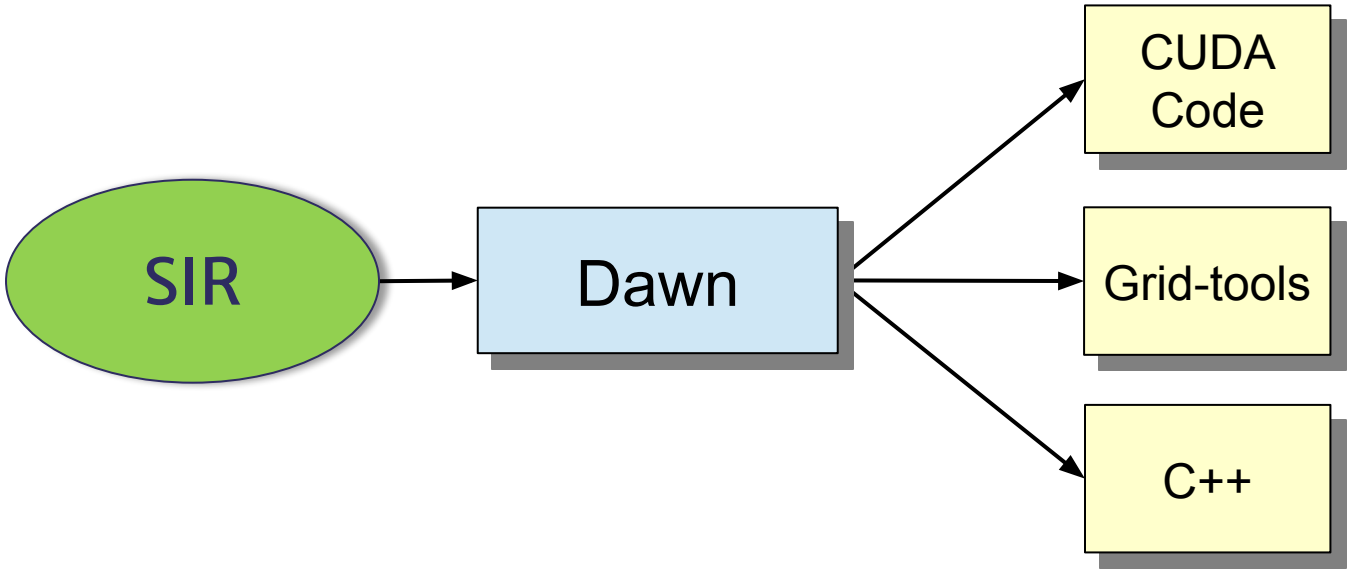
NEMO has structured mesh(es) with stencils



ESiWACE2 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823988

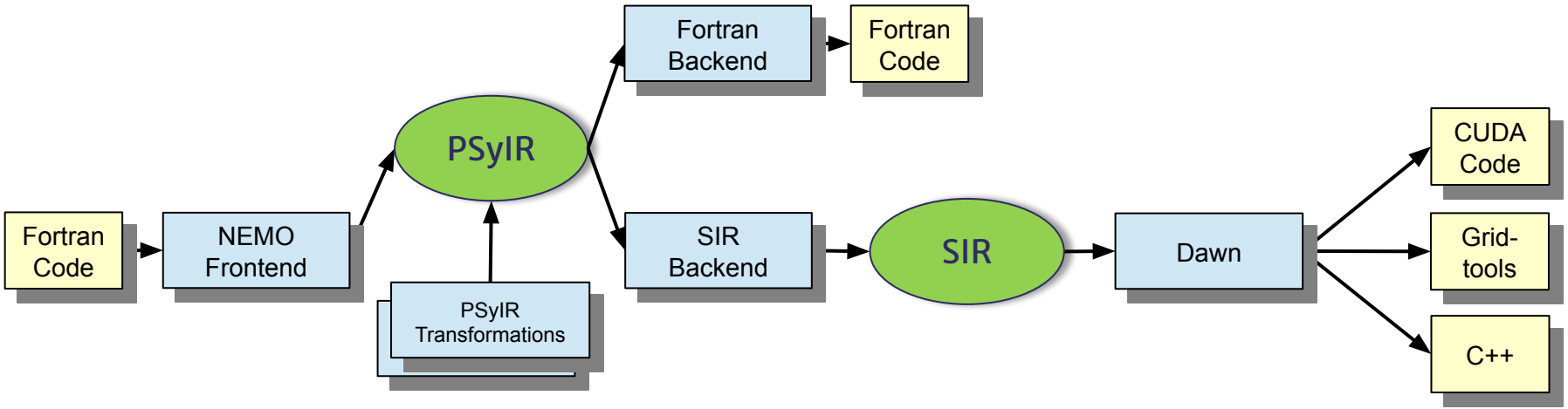
esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE

DAWN takes SIR as input



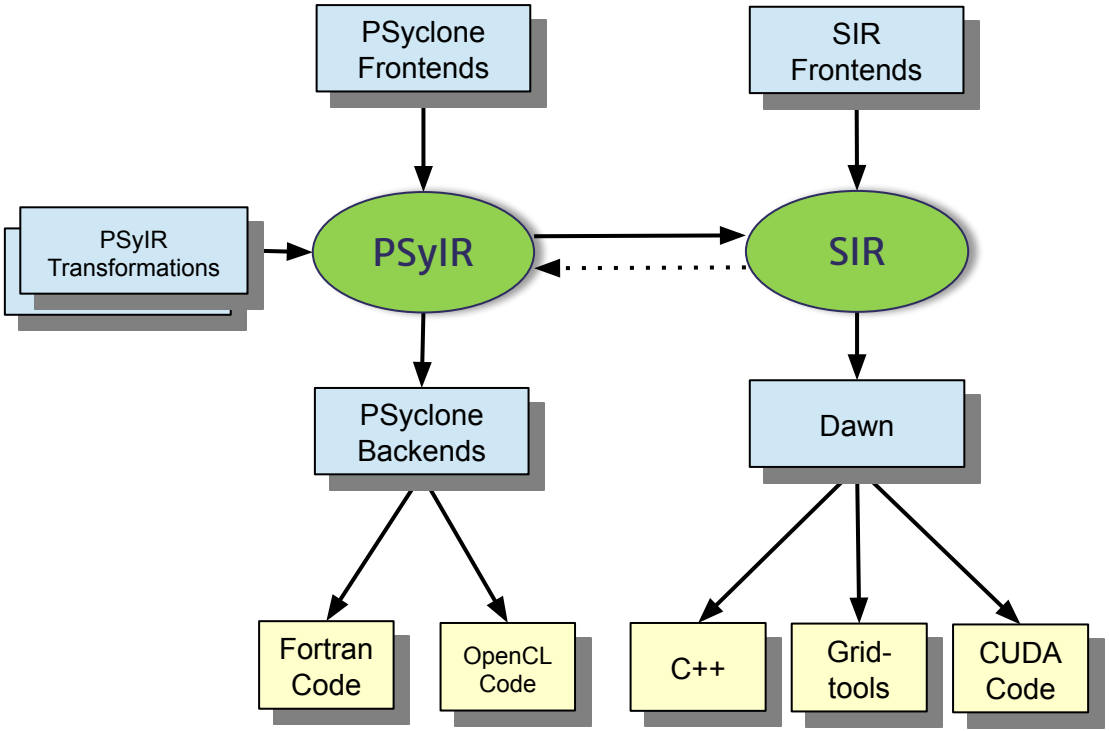
ESiWACE2 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823988

Joining these together



ESiWACE2 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823988

Alternative (better) view



ESiWACE2 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823988

SIR Python API and PSyIR

makeAssignmentStmt() — Assignment
makeFieldAccessExpr() — Array
makeBinaryOperator() — BinaryOperation
makeLiteralAccessExpr() — Literal (+UnaryOperation)

makeInterval()
makeAST()
makeVerticalRegionDeclStmt()
} NemoLoop

makeSIR()
makeStencil()
makeAST()
makeField()
} NemoInvokeSchedule

ESiWACE2 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823988

esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE

Fortran example

```
program hori_diff
  do k=1,n
     do j=1,n
        do i=1,n
           lap(i,j,k)=(-4.0)*in(i,j,k)+coeff(i,j,k)*( &
                      in(i+1,j,k)+in(i-1,j,k)+in(i,j+1,k)+in(i,j-1,k))
           out(i,j,k)=(-4.0)*lap(i,j,k)+coeff(i,j,k)*( &
                      lap(i+1,j,k)+lap(i-1,j,k)+lap(i,j+1,k)+lap(i,j-1,k))
        end do
     end do
  end do
end program hori_diff
```

ESiWACE2 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823988

PSyclone PSyIR representation (snippet)

```
NemoInvokeSchedule[invoke='hori_diff']
    0: Loop[type='unknown', field_space='None', it_space='None']
        Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
        Reference[name:'n']
        Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
        Schedule[]
            0: Loop[type='unknown', field_space='None', it_space='None']
                Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
                Reference[name:'n']
                Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
                Schedule[]
                    0: Loop[type='unknown', field_space='None', it_space='None']
                        Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
                        Reference[name:'n']
                        Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
                        Schedule[]
                            0: InlinedKern[]
                                Schedule[]
                                    0: Assignment[]
                                        ArrayReference[name:'lap']
                                            Reference[name:'i']
                                            Reference[name:'j']
                                            Reference[name:'k']
                                        BinaryOperation[operator:'ADD']
                                            BinaryOperation[operator:'MUL']
                                                UnaryOperation[operator:'MINUS']
                                                    Literal[value:'4.0', Scalar<REAL, UNDEFINED>]
                                                ArrayReference[name:'fin']
                                                    Reference[name:'i']
                                                    Reference[name:'j']
                                                    Reference[name:'k']
                                            BinaryOperation[operator:'MUL']
                                                ArrayReference[name:'coeff']
```

ESiWACE2 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823988

PSyclone generated SIR Python (snippet)

```
# PSyclone autogenerated SIR Python
verticalRegionFns = []
stencilname = "psyclone"
interval = makeInterval(Interval.Start, Interval.End, 0, 0)
bodyAST = makeAST([
  makeAssignmentStmt(
    makeFieldAccessExpr("lap",[0,0,0]),
    makeBinaryOperator(
      makeBinaryOperator(
        makeLiteralAccessExpr("-4.0", BuiltinType.Float),
        "*",
        makeFieldAccessExpr("in",[0,0,0])
        ),
      "+",
      makeBinaryOperator(
        makeFieldAccessExpr("coeff",[0,0,0]),
        "*",
        makeBinaryOperator(
          makeBinaryOperator(
            makeBinaryOperator(
              makeFieldAccessExpr("in",[1,0,0]),
              "+",
              makeFieldAccessExpr("in",[-1,0,0])
              ),
            "+",
            makeFieldAccessExpr("in",[0,1,0])
            ),
```

ESiWACE2 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823988

DAWN SIR representation

```
stencil psyclone
  field coeff['i', 'j', 'k'],out['i', 'j', 'k'],lap['i', 'j', 'k'],in['i', 'j', 'k'],
  vertical_region(kstart,kend)
    lap[0,0,0] = ((-4.0 * in[0,0,0]) + (coeff[0,0,0] * (((in[1,0,0] + in[-1,0,0]) + in[0,1,0]) + in[0,-1,0])))
    out[0,0,0] = ((-4.0 * lap[0,0,0]) + (coeff[0,0,0] * (((lap[1,0,0] + lap[-1,0,0]) + lap[0,1,0]) + lap[0,-1,0])))
```

ESiWACE2 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823988

DAWN cuda output (snippet)

```
  // initialized iterators
  int idx111 = (blockIdx.x*32+iblock)*1+(blockIdx.y*4+jblock)*stride_111_1;

  // jump iterators to match the intersection of beginning of next interval and the parallel execution block
  idx111 += max(0, blockIdx.z * 4) * stride_111_2;
  int kleg_lower_bound = max(0,blockIdx.z*4);
  int kleg_upper_bound = min( ksize - 1 + 0,(blockIdx.z+1)*4-1);;
for(int k = kleg_lower_bound+0; k <= kleg_upper_bound+0; ++k) {
  if(iblock >= -1 && iblock <= block_size_i -1 + 1 && jblock >= -1 && jblock <= block_size_j -1 + 1) {
lap[idx111] = (((gridtools::clang::float_type) -4.0 * __ldg(&(in[idx111])) + (__ldg(&(coeff[idx111])) * (((__ldg(&
(in[idx111+1*1])) + __ldg(&(in[idx111+1*-1]))) + __ldg(&(in[idx111+stride_111_1*1]))) + __ldg(&(in[idx111+stride_11
1_1*-1])))));
  }    __syncthreads();
  if(iblock >= 0 && iblock <= block_size_i -1 + 0 && jblock >= 0 && jblock <= block_size_j -1 + 0) {
out[idx111] = (((gridtools::clang::float_type) -4.0 * lap[idx111]) + (__ldg(&(coeff[idx111])) * (((lap[idx111+1*1]
+ lap[idx111+1*-1]) + lap[idx111+stride_111_1*1]) + lap[idx111+stride_111_1*-1])));
  }
    // Slide kcaches

    // increment iterators
    idx111+=stride_111_2;
}}
```

ESiWACE2 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823988

Working towards ...

NEMO tracer advection benchmark

`<psyclone_home>/examples/nemo/eg4`

Commented out the bits that are not working

- 1D and 2D arrays
- Implicit loops
- Imperfectly nested loops
- Issue with `... = - X ...`

PSyclone replaces Fortran intrinsics with code

- `SIGN`, `MIN`, `ABS`

Transforms 183 lines of Fortran generating 1132 lines of Python SIR

ESiWACE2 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823988

Summary

- PSyIR nemo api maps well to SIR
- It is possible to translate from PSyIR to SIR (where SIR supports the PSyIR)
- Simple examples work
- PSyclone can transform the code to allow translation
- Working on nemo tracer advection benchmark

ESiWACE2 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 823988

Science and Technology Facilities Council

Thank you

Facebook: Science and Technology Facilities Council

Twitter:@STFC_matters

YouTube: Science and Technology Facilities Council