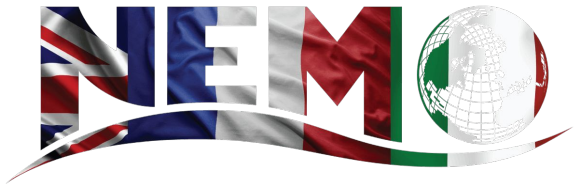
Science and Technology Facilities Council

Overview of Using PSyclone with

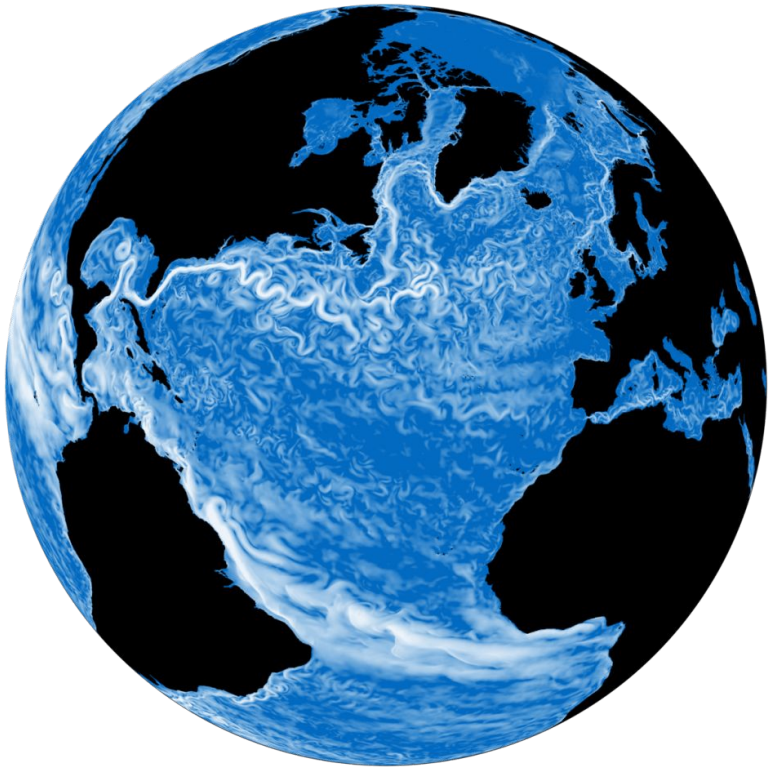


Andy Porter, STFC Hartree Centre

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

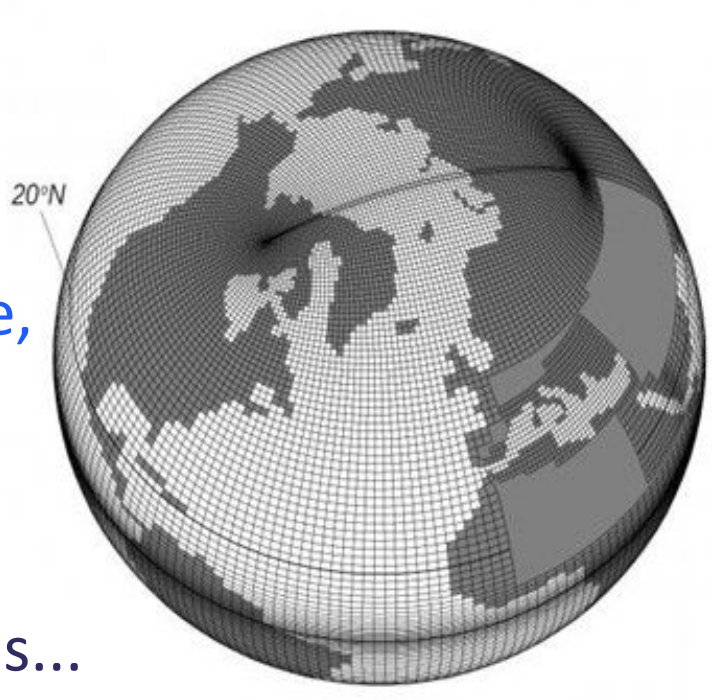
Outline

- The NEMO Ocean Model
- Performance characteristics
- Coding standards
- Using PSyclone with NEMO
- Existing code => PSyIR, navigating the PSyIR
- The tracer-advection mini-app



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

The NEMO Ocean Model



- Finite difference model using a tripolar, stretched latitude, longitude mesh ('ORCA')
- Three core components:
  - NEMO-OPA: ocean dynamics, thermodynamics
  - NEMO-SI^3: sea-ice (thermo)dynamics, brine inclusions...
  - NEMO-TOP/PISCES: tracer transport and biogeochemistry
- Mesh rotated so that poles are over land
  - Can go to high resolution without the 'pole problem'
  - No need to fundamentally change the code
- Relatively large (core of ~100K lines of Fortran)

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

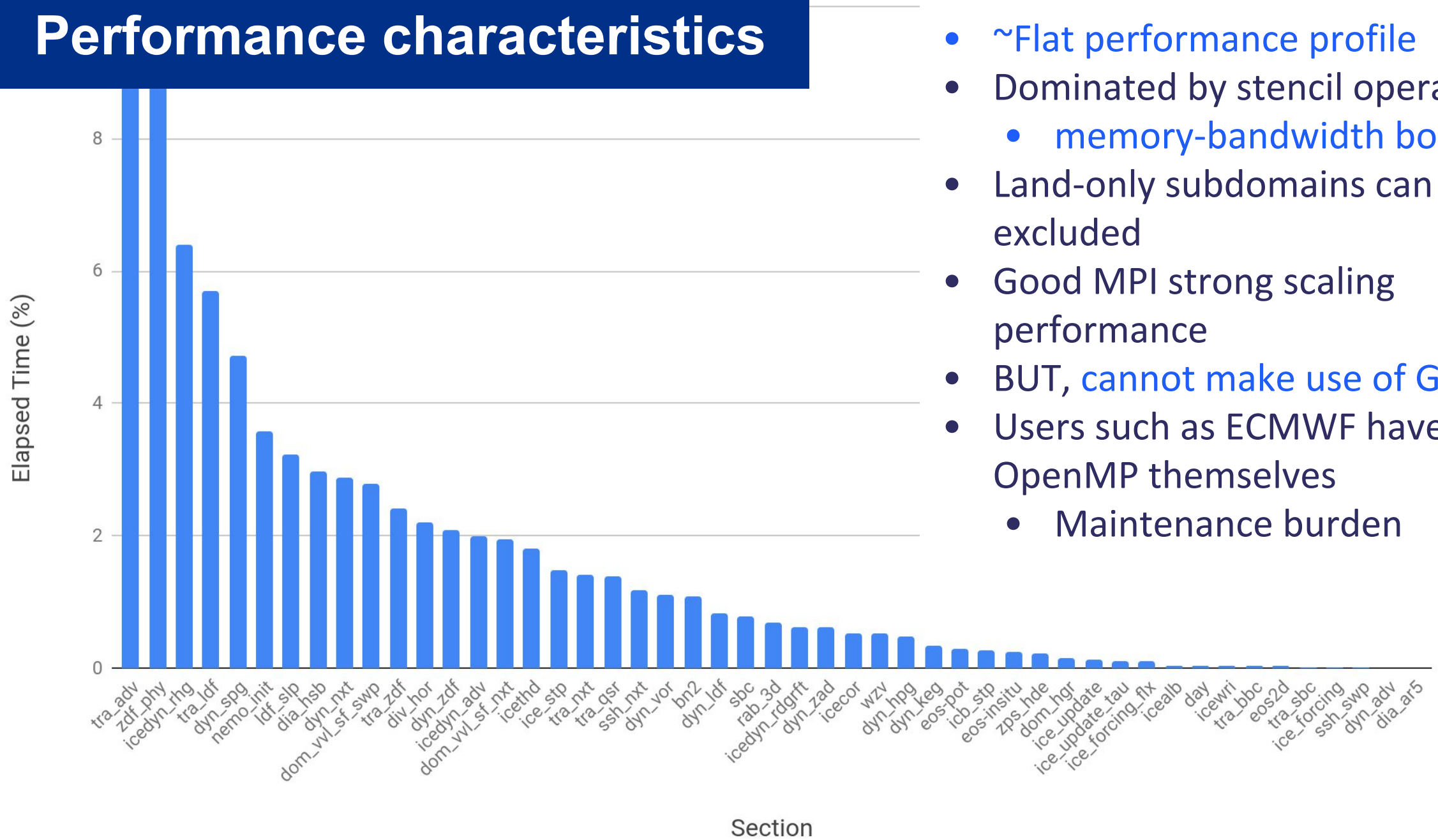
NEMO continued...

- Large user base (Met Office, ECMWF, CERFACS...)
  - Trusted part of CMIP experiments
- Constantly under scientific development
- Developers are typically oceanographers, not computer scientists
- ~20 years old (and counting...)
- Focus on:
  - Scientific correctness
  - Maintainability (aka Productivity)
  - Portability
- Run either in serial or MPI parallel
  - => CPU only

```
121 ./ICE/icetab.F90
317 ./ICE/icesbc.F90
264 ./ICE/icerst.F90
1649 ./ICE/icedyn_adv_umx.F90
166 ./ICE/icethd_sal.F90
573 ./ICE/icethd.F90
483 ./ICE/ice.F90
251 ./ICE/icethd_pnd.F90
755 ./ICE/iceitd.F90
886 ./ICE/icethd_zdf_bl99.F90
461 ./ICE/iceupdate.F90
128 ./ICE/icethd_zdf.F90
694 ./ICE/icectl.F90
205 ./ICE/icethd_da.F90
188 ./ICE/icecor.F90
169 ./ICE/icedyn_adv.F90
144 ./ICE/icethd_ent.F90
856 ./ICE/icedyn_adv_pra.F90
1048 ./ICE/icevar.F90
442 ./SAO/nemogcm.F90
1998 ./SAO/obs_fbm.F90
81 ./SAO/sao_data.F90
58 ./SAO/sao_intp.F90
173 ./SAO/sao_read.F90
88 ./TOP/TRP/trczdf.F90
235 ./TOP/TRP/trcsink.F90
195 ./TOP/TRP/trcldf.F90
109 ./TOP/TRP/trctrp.F90
295 ./TOP/TRP/trcrad.F90
215 ./TOP/TRP/trcsbc.F90
279 ./TOP/TRP/trcnxt.F90
991 ./TOP/TRP/trdmxl_trc.F90
125 ./TOP/TRP/trdtrc.F90
243 ./TOP/TRP/trcadv.F90
102 ./TOP/TRP/trcbbl.F90
163 ./TOP/TRP/trdtrc_oce.F90
386 ./TOP/TRP/trcdmp.F90
210 ./TOP/TRP/trdmxl_trc_rst.F90
597 ./TOP/trcsub.F90
203 ./TOP/PISCES/P2Z/p2zopt.F90
70 ./TOP/PISCES/P2Z/p2zsms.F90
492 ./TOP/PISCES/P2Z/p2zbio.F90
154 ./TOP/PISCES/P2Z/p2zsed.F90
255 ./TOP/PISCES/P2Z/p2zexp.F90
70 ./TOP/PISCES/par_pisces.F90
194 ./TOP/PISCES/sms_pisces.F90
93 ./TOP/PISCES/trcnam_pisces.F90
43 ./TOP/PISCES/SED/sedmodel.F90
138 ./TOP/PISCES/SED/sedwri.F90
257 ./TOP/PISCES/SED/sedmbc.F90
56 ./TOP/PISCES/SED/oce_sed.F90
36 ./TOP/PISCES/SED/sed_oce.F90
97 ./TOP/PISCES/SED/sedstp.F90
```

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

Performance characteristics



- ~Flat performance profile
- Dominated by stencil operations
  - memory-bandwidth bound
- Land-only subdomains can be excluded
- Good MPI strong scaling performance
- BUT, cannot make use of GPUs
- Users such as ECMWF have added OpenMP themselves
  - Maintenance burden

```
zwx(:,:,jpk) = 0.e0    ;    zwy(:,:,jpk) = 0.e0

DO jk = 1, jpk-1
   DO jj = 1, jpj-1
      DO ji = 1, jpi-1
         zwx(ji,jj,jk) = umask(ji,jj,jk) * ( mydomain(ji+1,jj
         zwy(ji,jj,jk) = vmask(ji,jj,jk) * ( mydomain(ji,jj+1
      END DO
   END DO
END DO

zslpx(:,:,jpk) = 0.e0    ;    zslpy(:,:,jpk) = 0.e0

DO jk = 1, jpk-1
   DO jj = 2, jpj
      DO ji = 2, jpi
         zslpx(ji,jj,jk) =                          ( zwx(ji,jj,jk)
         &              * ( 0.25d0 + SIGN( 0.25d0, zwx(ji,jj,jk)
         zslpy(ji,jj,jk) =                          ( zwy(ji,jj,jk)
         &              * ( 0.25d0 + SIGN( 0.25d0, zwy(ji,jj,jk)
      END DO
   END DO
END DO

DO jk = 1, jpk-1
   DO jj = 2, jpj
      DO ji = 2, jpi
         zslpx(ji,jj,jk) = SIGN( 1.d0, zslpx(ji,jj,jk) ) * MIN(    ABS( zslpx(ji  ,jj,jk) ),   &
         &                                                          2.d0*ABS( zwx  (ji-1,jj,jk) ),   &
         &                                                          2.d0*ABS( zwx  (ji  ,jj,jk) ) )
         zslpy(ji,jj,jk) = SIGN( 1.d0, zslpy(ji,jj,jk) ) * MIN(    ABS( zslpy(ji,jj  ,jk) ),   &
         &                                                          2.d0*ABS( zwy  (ji,jj-1,jk) ),   &
         &                                                          2.d0*ABS( zwy  (ji,jj  ,jk) ) )
      END DO
   END DO
END DO
```

Large code base, but:

- Fairly homogeneous (stencils)
- Apply same/similar optimisations everywhere
- Strict coding standards
- Have domain-specific knowledge, e.g.:
  - Loops over longitude, latitude and vertical levels
  - Tracers
  - Run-time constants
  - Temporary variables

DSL front-ends

Unmodified existing code

Constrained existing code

High-level domain-specific language

Unconstrained environment

Constrained environment

Performance limitations

High Performance

Supports evolution

Requires revolution

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

DSL front-ends

NEMO

LFRic

High-level domain-specific language

Unconstrained environment

Constrained environment

Performance limitations

High Performance

Supports evolution

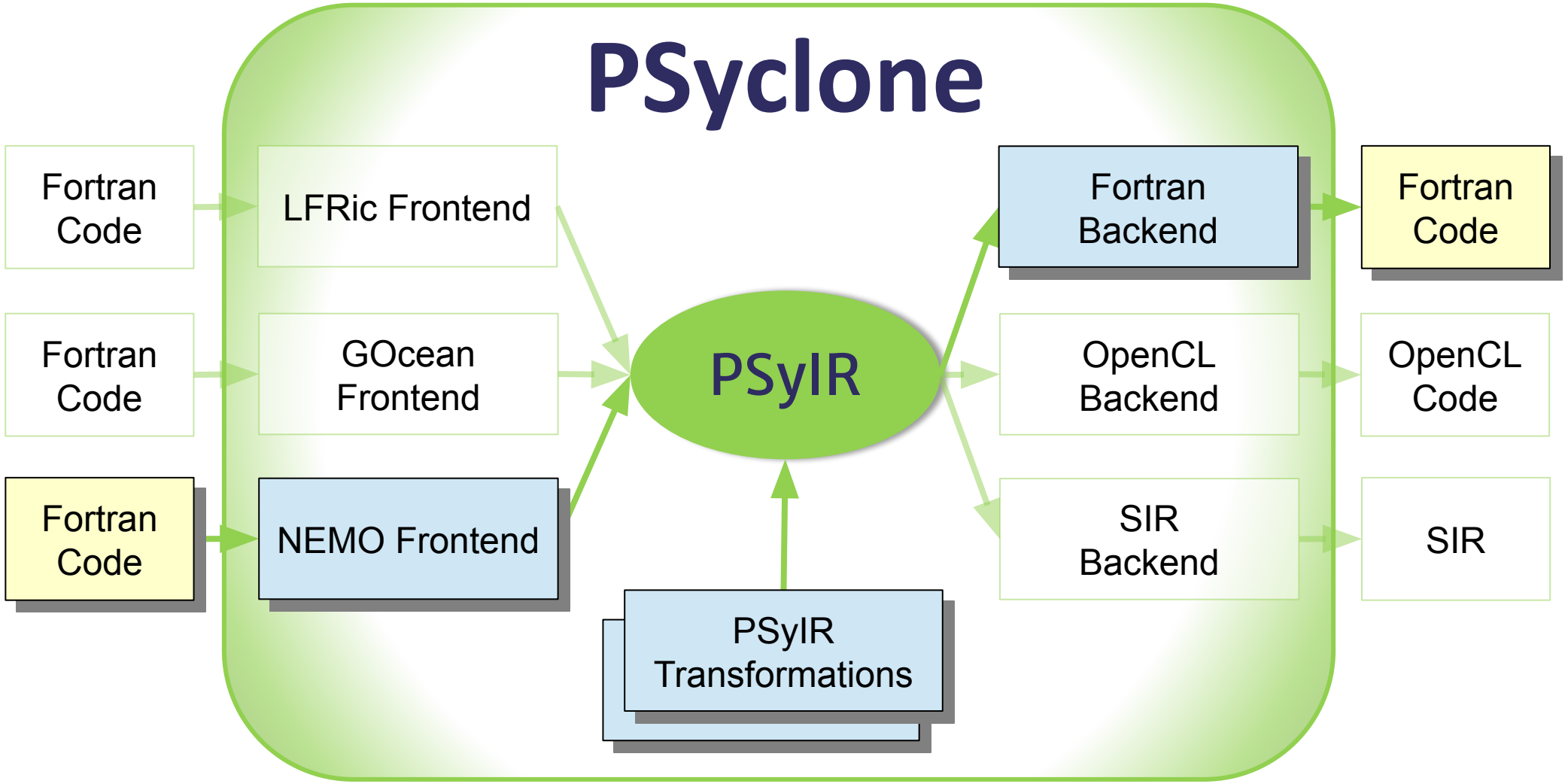
Requires revolution

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

Strategy

- Jeremy Appleyard (NVIDIA) previously obtained good performance using just OpenACC KERNELS (v3.4 of NEMO)
- Extend PSyclone such that we can replicate his results
  - without manual code modification (ideally)
  - on ***any*** NEMO configuration
- Use NVIDIA's 'managed-memory' option to deal with data movement between CPU/GPU
- Extend PSyclone with functionality to introduce more tailored OpenACC directives as required

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



(PSyIR: PSyclone Internal Representation)

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

Understanding the PSyIR of NEMO code

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

PSyIR for array assignments (implicit loops)

```
zwx(:,:,jpk) = 0.e0    ;    zwy(:,:,jpk) = 0.e0

DO jk = 1, jpk-1
   DO jj = 1, jpj-1
      DO ji = 1, jpi-1
         zwx(ji,jj,jk) = umask(ji,jj,jk) * ( mydomain(ji+1,jj,jk) - mydomain(ji,jj,jk) )
         zwy(ji,jj,jk) = vmask(ji,jj,jk) * ( mydomain(ji,jj+1,jk) - mydomain(ji,jj,jk) )
      END DO
   END DO
END DO

zslpx(:,:,jpk) = 0.e0    ;    zslpy(:,:,

DO jk = 1, jpk-1
   DO jj = 2, jpj
      DO ji = 2, jpi
         zslpx(ji,jj,jk) =
         &                 * ( 0.25d0 + SIGN
         zslpy(ji,jj,jk) =
         &                 * ( 0.25d0 + SIGN
      END DO
   END DO
END DO

DO jk = 1, jpk-1
   DO jj = 2, jpj
      DO ji = 2, jpi
         zslpx(ji,jj,jk) = SIGN( 1.d0,
         &
         &
         zslpy(ji,jj,jk) = SIGN( 1.d0,
         &
         &
      END DO
   END DO
END DO
```

```
1: Assignment[]
    ArrayReference[name:'zwx']
        Range[]
            BinaryOperation[operator:'LBOUND']
                Reference[name:'zwx']
                Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
            BinaryOperation[operator:'UBOUND']
                Reference[name:'zwx']
                Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
            Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
        Range[]
            BinaryOperation[operator:'LBOUND']
                Reference[name:'zwx']
                Literal[value:'2', Scalar<INTEGER, UNDEFINED>]
            BinaryOperation[operator:'UBOUND']
                Reference[name:'zwx']
                Literal[value:'2', Scalar<INTEGER, UNDEFINED>]
            Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
        Reference[name:'jpk']
    Literal[value:'0.e0', Scalar<REAL, SINGLE>]
2: Assignment[]
    ArrayReference[name:'zwy']
        Range[]
```

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

PSyIR for explicit loops

```
zwx(:,:,jpk) = 0.e0    ;    zwy(:,:,jpk) = 0.e0

DO jk = 1, jpk-1
   DO jj = 1, jpj-1
      DO ji = 1, jpi-1
         zwx(ji,jj,jk) = umask(ji,jj,jk) * ( mydomain(ji+1,jj,jk) - mydomain(ji,jj,jk) )
         zwy(ji,jj,jk) = vmask(ji,jj,jk) * ( mydomain(ji,jj+1,jk) - mydomain(ji,jj,jk) )
      END DO
   END DO
END DO

zslpx(:,:,jpk) = 0.e0    ;    zslpy(:,:,jpk) = 0.e0

DO jk = 1, jpk-1
   DO jj = 2, jpj
      DO ji = 2, jpi
         zslpx(ji,jj,jk) =                          ( zwx(
         &               * ( 0.25d0 + SIGN( 0.25d0, zwx
         zslpy(ji,jj,jk) =                          ( zwy(
         &               * ( 0.25d0 + SIGN( 0.25d0, zwy
      END DO
   END DO
END DO

DO jk = 1, jpk-1
   DO jj = 2, jpj
      DO ji = 2, jpi
         zslpx(ji,jj,jk) = SIGN( 1.d0, zslpx(ji,jj,j
         &
         &
         zslpy(ji,jj,jk) = SIGN( 1.d0, zslpy(ji,jj,j
         &
         &
      END DO
   END DO
END DO
```

```
        Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
    Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
    Schedule[]
        0: Loop[type='lat', field_space='None', it_space='None']
            Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
            BinaryOperation[operator:'SUB']
                Reference[name:'jpj']
                Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
            Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
            Schedule[]
                0: Loop[type='lon', field_space='None', it_space='None']
                    Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
                    BinaryOperation[operator:'SUB']
                        Reference[name:'jpi']
                        Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
                    Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
                    Schedule[]
                        0: InlinedKern[]
                            Schedule[]
                                0: Assignment[]
                                    ArrayReference[name:'zwx']
                                        Reference[name:'ji']
                                        Reference[name:'jj']
                                        Reference[name:'jk']
                                    BinaryOperation[operator:'MUL']
                                        ArrayReference[name:'umask']
                                            Reference[name:'ji']
                                            Reference[name:'jj']
                                            Reference[name:'jk']
                                        BinaryOperation[operator:'SUB']
                                            ArrayReference[name:'mydomain']
                                                BinaryOperation[operator:'ADD']
                                                    Reference[name:'ji']
                                                    Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
                                                Reference[name:'jj']
                                                Reference[name:'jk']
```

ESiWACE2 has received funding from the European Un

PSyIR with CodeBlock

```
OPEN(unit = 4, file = 'output.dat', form='formatted')

DO jk = 1, jpk-1
   DO jj = 2, jpj-1
      DO ji = 2, jpi-1
         write(4,*) mydomain(ji,jj,jk)
      END DO
   END DO
END DO

CLOSE(4)
```

```
13: Loop[type='levels', field_space='None', it_space='None']
    Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
    BinaryOperation[operator:'SUB']
        Reference[name:'jpk']
        Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
    Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
    Schedule[]
        0: Loop[type='lat', field_space='None', it_space='None']
            Literal[value:'2', Scalar<INTEGER, UNDEFINED>]
            BinaryOperation[operator:'SUB']
                Reference[name:'jpj']
                Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
            Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
            Schedule[]
                0: Loop[type='lon', field_space='None', it_space='None']
                    Literal[value:'2', Scalar<INTEGER, UNDEFINED>]
                    BinaryOperation[operator:'SUB']
                        Reference[name:'jpi']
                        Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
                    Literal[value:'1', Scalar<INTEGER, UNDEFINED>]
                    Schedule[]
                        0: CodeBlock[[<class 'fparser.two.Fortran2003.Write_Stmt'>]]
```

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

Tracer-advection mini-app

- Hands-on sessions will all use the Tracer-advection mini-app
  - Developed by CMCC
  - Basis of the 'NEMO Dwarf' (ESCAPE Project)
- Based on a tracer-advection routine from NEMO
  - Always high in performance profile (see earlier)
- Outer loop over tracer species becomes an 'iteration' loop
- Representative of majority of code base
- No MPI
- 288 lines of Fortran
- No inputs required - generates synthetic data

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

Hands-on and questions...

```
$ git clone --recursive https://github.com/stfc/PSyclone.git
$ cd PSyclone
$ pip install .
$ cd tutorials/practicals/nemo/1_nemo_psyir
```

https://github.com/stfc/PSyclone/tree/master/tutorial/practicals/nemo/1_nemo_psyir

Questions on Slack: https://join.slack.com/t/meteoswiss-group/shared_invite/zt-j7ry1st0-4TW0D9B_auq7tDa4zylrqQ

Andy Porter
andrew.porter@stfc.ac.uk

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