Advanced Computation and I/O Methods for Earth-System Simulations

Julian M. Kunkel, Thomas Ludwig, Thomas Dubos, Naoya Maruyama, Takayuki Aoki, Günther Zängl, Hisashi Yashiro, Ryuji Yoshida, Hirofumi Tomita, Masaki Satoh, Yann Meurdesoif

Scientific Computing
Department of Informatics
University of Hamburg

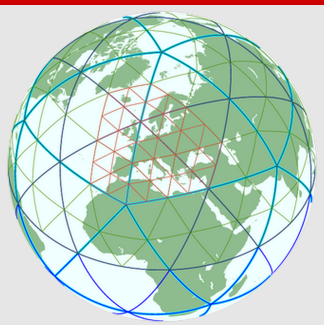
2016-01-27

Goals

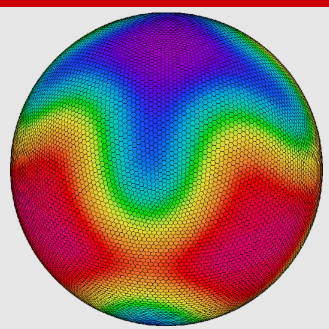
Address key issues of icosahedral earth-system models

- Enhance programmability and performance-portability
- Overcome storage limitations
- Additional benefit: a common benchmark for these models

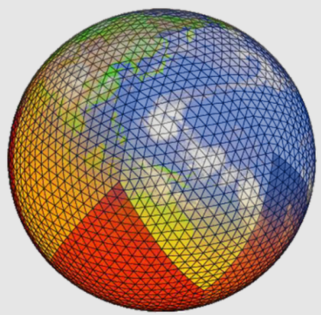
Covered models



ICON

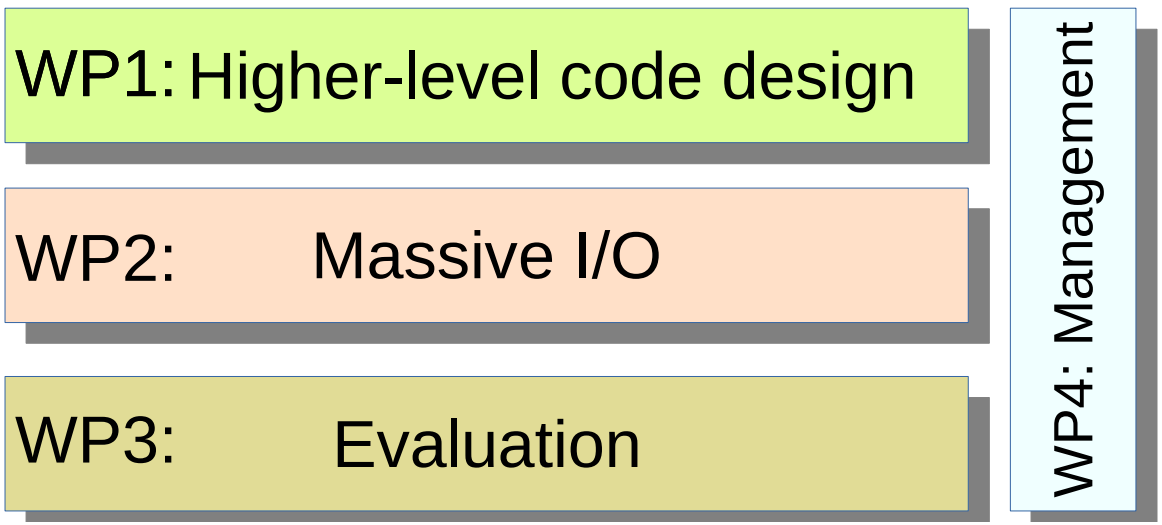


DYNAMICO



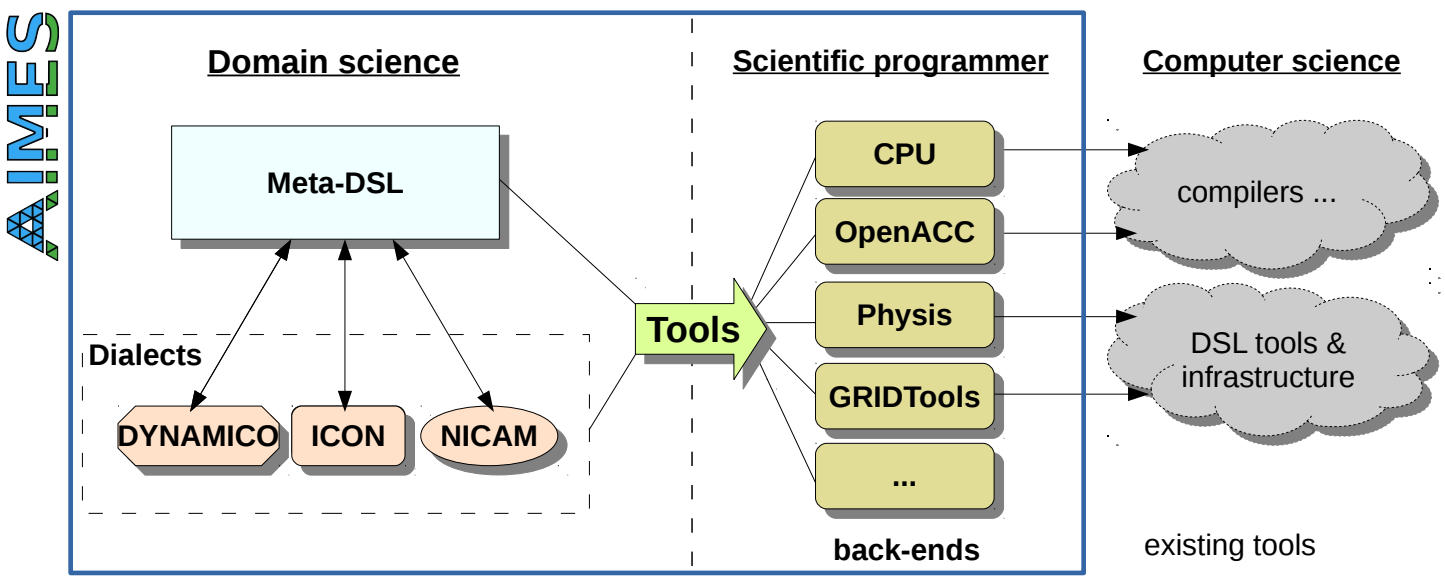
NICAM

Work-Package Overview



Objectives: WP 1 Towards Higher-Level Code Design

- Separation of concerns: Domain science, scientific programmer, CS
  - High level of abstraction, close to application domain
  - Independence of hardware-specific features, e.g. memory-layout
  - Convertible into existing languages and DSLs



WP 1: Tasks & Strategy

- Develop and reformulate key parts of models into DSL-dialects
  - DYNAMICO
  - ICON
  - NICAM
- Design a common DSL concepts for all ICO models
  - Identify chances for common convention
  - But still allow for individual adjustments/formulations
- Develop a source-to-source translation tool and mappings
  - Lightweight to allow scientists not to worry about maintenance
  - Language can be adjusted for individual models / groups

Example: Prototypical Work for ICON

Dialect for ICON

```
Subset                      :: p_patch
Edge                        :: edge
REAL(wp), Var1DCell         :: z_thermal_exp
foreach cell in p_patch (elevation rl_start to rl_end)
  z_thermal_exp(cell) = z_thermal_exp(cell) + &
    cvd_o_rd * p_nh%diag%ddt_exner_phy(cell) /  &
    (p_nh%prog(nnow)%exner(cell)* p_nh%metrics%inv_ddqz_z_full(cell))
end foreach
```

Original code == Translated code for CPUs

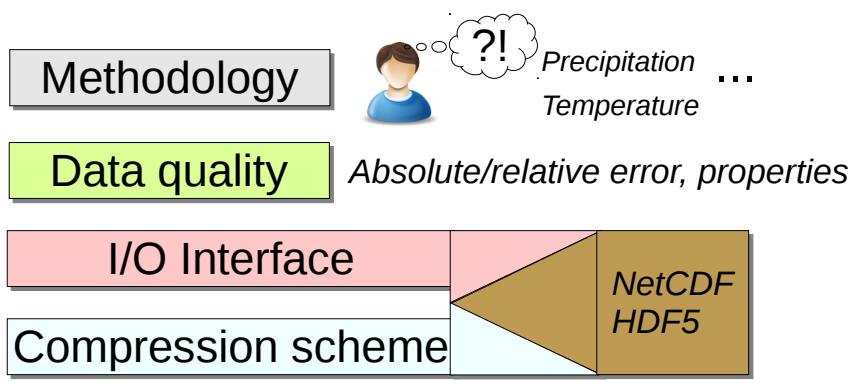
```
TYPE(t_patch), TARGET, INTENT(IN) :: p_patch
REAL(wp):: z_thermal_exp (nproma,p_patch%nblks_c)
i_startblk = p_patch%cells%start_block(rl_start)
i_endblk   = p_patch%cells%end_block(rl_end)
DO jb = i_startblk, i_endblk
  CALL get_indices_c(p_patch, jb, i_startblk, i_endblk,&
       i_startidx, i_endidx, rl_start, rl_end)
  DO jk = 1, nlev
    DO jc = i_startidx, i_endidx
      z_thermal_exp(jc,jb) = z_thermal_exp(jc,jb) +     &
         cvd_o_rd * p_nh%diag%ddt_exner_phy(jc,jk,jb)/ &
         (p_nh%prog(nnow)%exner(jc,jk,jb)*             &
         p_nh%metrics%inv_ddqz_z_full(jc,jk,jb))
    ENDDO
  ENDDO
ENDDO
```

Example: Physis DSL a Target for the Translation

```
void kernel(const int x, const int y, const int z,
            PSGrid3DPoint g1, PSGrid3DDouble g2) {
  double v = PSGridGet(g1,x,y,z).vx
    +PSGridGet(g1,x-1,y,z).vx + PSGridGet(g1,x+1,y,z).vx
    +PSGridGet(g1,x,y-1,z).vy + PSGridGet(g1,x,y+1,z).vy
    +PSGridGet(g1,x,y,z-1).vz + PSGridGet(g1,x,y,z+1).vz;
  PSGridEmit(g2,v/7.0);
}

...

PSGrid3DPoint g1 = PSGrid3DPointNew(NX, NY, NZ);
PSGrid3DDouble g2 = PSGrid3DDoubleNew(NX, NY, NZ);
PSDomain3D d = PSDomain3DNew(0, NX, 0, NY, 0, NZ);

PSStencilMap(kernel,d,g1,g2);
double max_value;
PSReduce(&max_value, g2, PS_MAX);
```

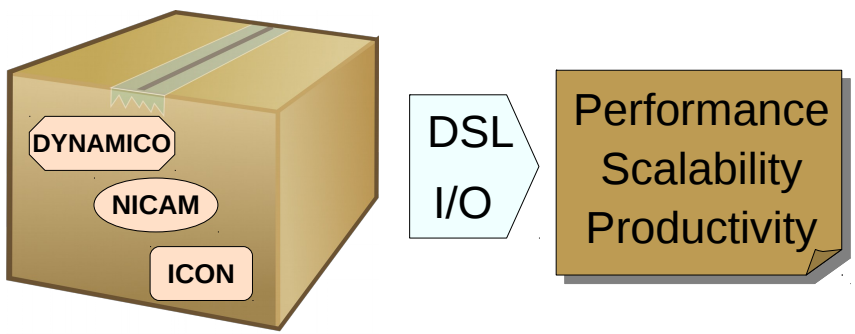
Objectives WP2: Massive I/O

- Optimization of I/O middleware for icosahedral data
  - Throughput, metadata handling
- Design of domain-specific compression (c. ratio $> 10 : 1$)
  - Investigate metrics allowing to define accuracy per variable
  - User-interfaces for specifying accuracy
  - Methodology for identifying the required accuracy
  - Compression schemes exploiting this knowledge



Objectives WP 3: Evaluation

- Evaluating the DSL and domain-specific I/O advancements
- Providing a common benchmark package for all models



Partners and Expertise

Funded partners


Thomas Ludwig (Universität Hamburg)
I/O middleware, compression, ICON DSL


Thomas Dubos (Institut Pierre Simon Laplace)
Application I/O servers, compression, DYNAMICO


Naoya Maruyama (RIKEN)
DSL (Physis), GPUs, NICAM


Takayuki Aoki (Tokio Institute of Technology)
DSL (HybridFortran), language extension, peta-scale apps

Cooperation Partners

- DKRZ *(I/O, DSL)*
- DWD *(ICON, DSL, I/O)*
- University of Exeter *(Math. aspects in the DSL)*
- CSCS *(GPU/ICON, GRIDTool, compression)*
- Intel *(DSL-backend optimization for XeonPhi, CPU)*
- NVIDIA *(DSL-backend optimization for GPU)*
- The HDF Group *(I/O, unstructured data, compression)*
- NCAR (MPAS developers, another icosahedral model)
- Bull
- Cray

Information exchange, participate in workshops, [hardware access]

Perspectives in SPPEXA

- Effort towards standardization
  - Common Meta-DSL for icosahedral models
  - I/O interfaces/schemes for compression
- Applicability beyond the scientific domain of climate/NWP
  - Advancement of I/O middleware are feed back to communities
  - DSL tools and strategies can be used for other domains
- Support for early researchers
  - We offer research stays
- We would be happy to collaborate with you

Summary

- AIMES covers programmability issues on the high-level
  - DSL-extensions enrich existing languages
  - Fosters separation of concerns, increase performance portability
- AIMES addresses domain-specific lossy compression
  - (Help) scientists to define the variable accuracy
  - Exploit this knowledge in the compression scheme
- AIMES works towards a common benchmark suite
  - Reduce hurdle for computer scientists to experiment with models