Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Home Affairs FDHA
Federal Office of Meteorology and Climatology MeteoSwiss

Developing a km-scale model system over the Alpine arc

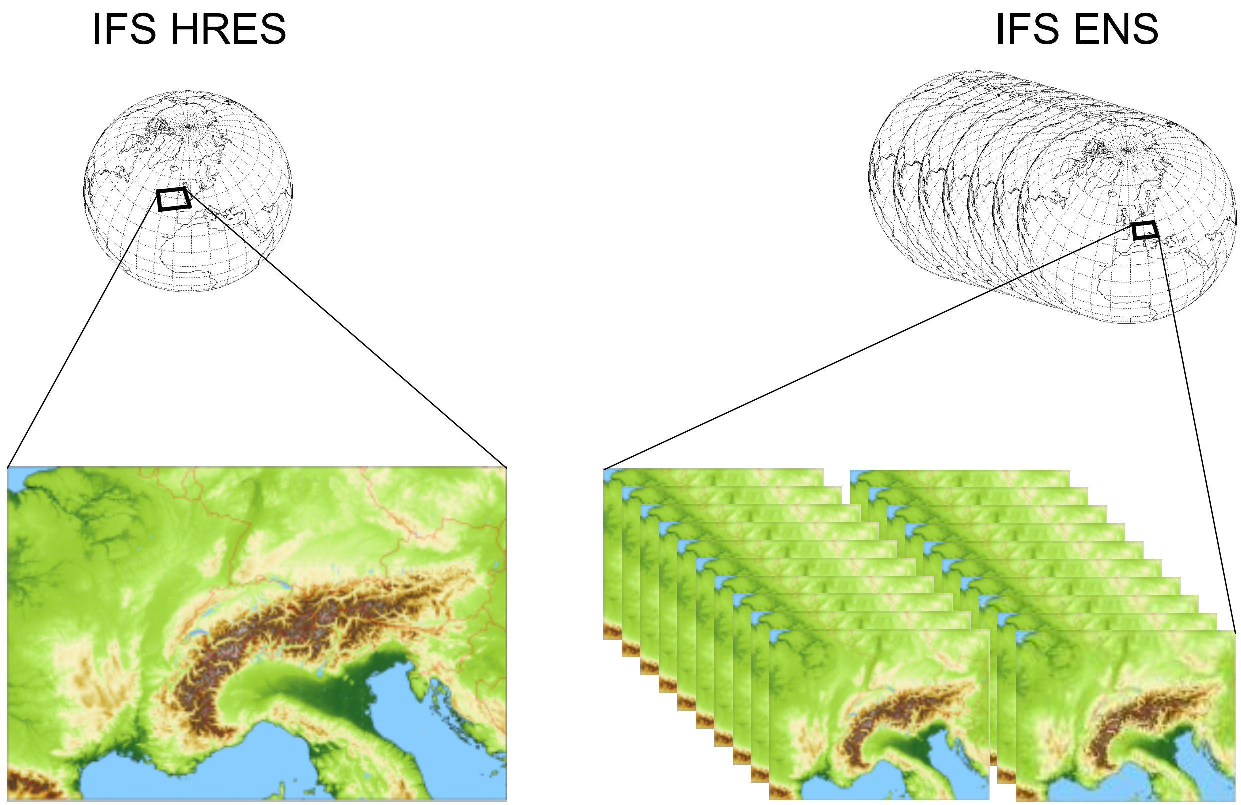
HPC and other challenges

Oliver Fuhrer[1], *M. Arpagaus*[1], *J.-M. Bettems*[1], *S. Böing*[7], *B. Goger*[8], *T. Gysi*[3], *D. Leuenberger*[1], *X. Lapillonne*[1], *G. de Morsier*[1], *C. Osuna*[1], *M. Rotach*[8], *J. Schmidli*[3], *P. Steiner*[1], *T. Schulthess*[5,6] , *A. Walser*[1], *et al.*

[1]*Federal Institute of Meteorology and Climatology, MeteoSwiss*
[2]*ITS Research Informatics, ETH Zurich*
[3]*University of Frankfurt*
[4]*Institute for Atmospheric and Climate Science, ETH Zurich 5Institute for Theoretical Physics, ETH Zurich*
[6]*Swiss National Supercomputing Centre, CSCS*
[7]*University of Leeds*
[7]*University of Innsbruck*

Greyzone Workshop, ECMWF, November 13-16, 2017

Operational system since 2016



COSMO-1

1.1 km gridspacing
8 x per day
+33h forecast
deterministic

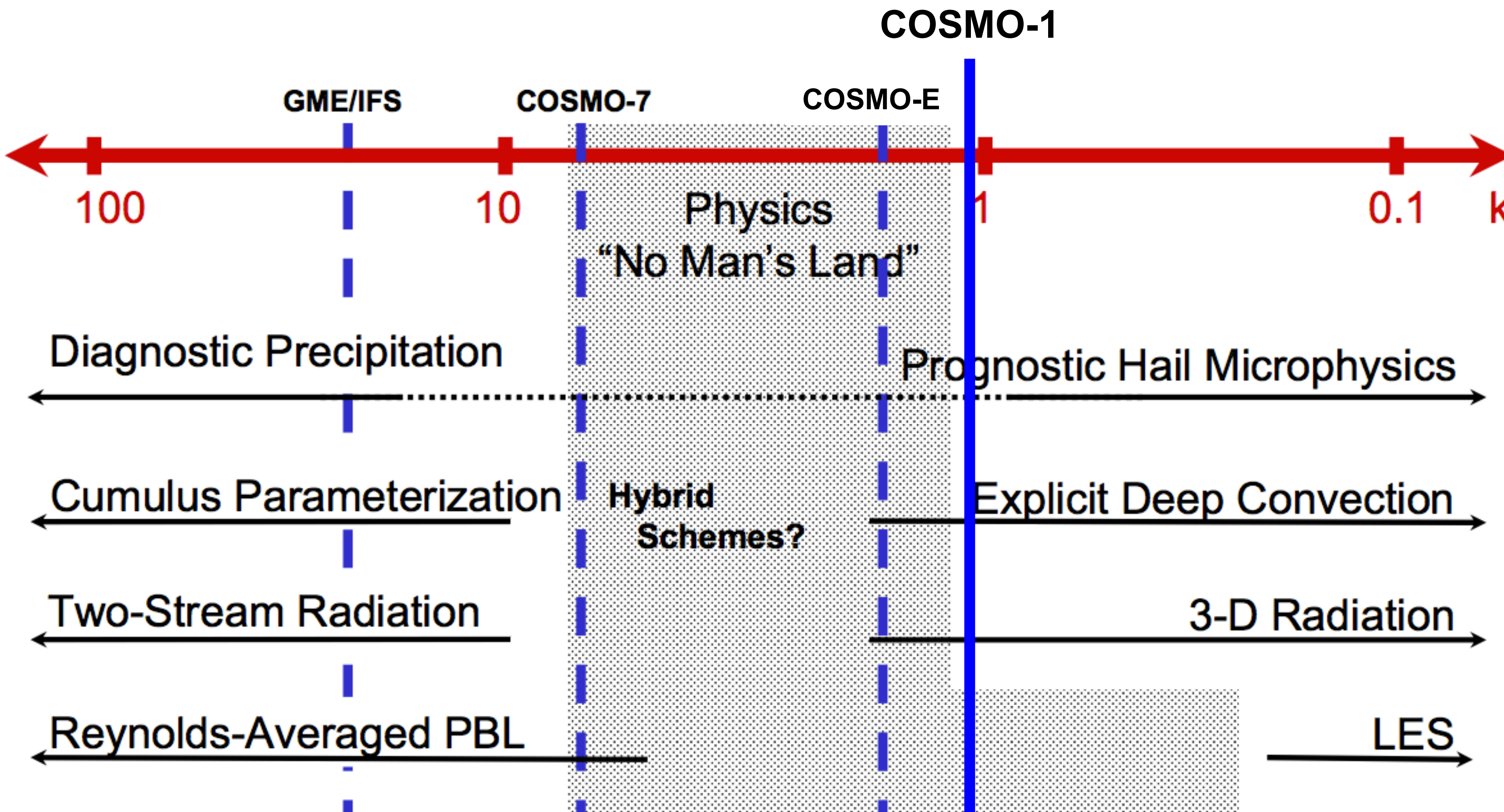
COSMO-E

2.2 km gridspacing
2 x per day
+5d forecast
21 members

Ensemble data assimilation: LETKF (40 members)

Greyzone

(adapted from Klemp 2007)



COSMO-1 can steer clear of a large part of the "grey zone" by jumping ahead to Δx = 1.1 km

The challenge of NWP in Switzerland

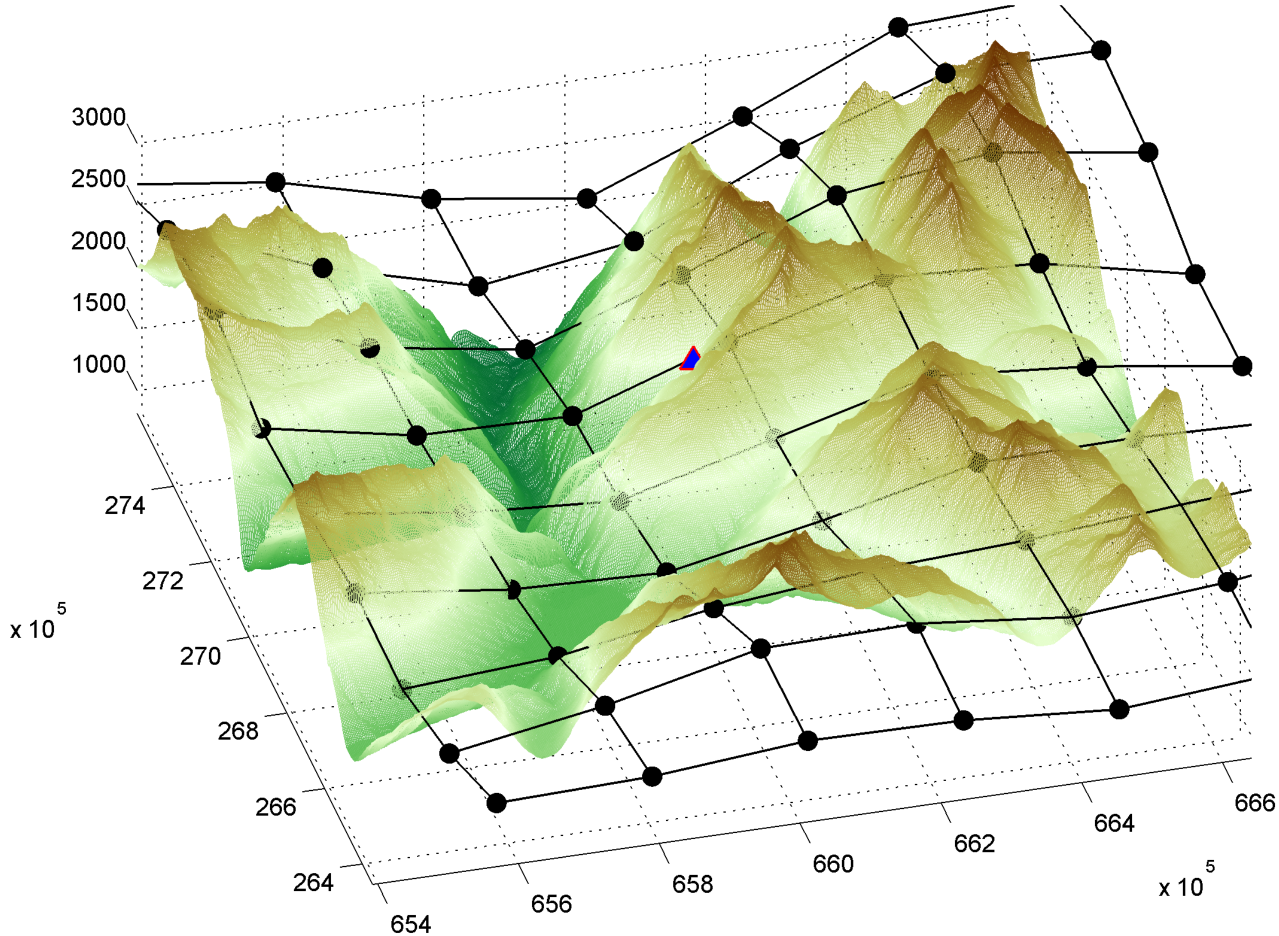
Flat-terrain cloudy BL

Cloudy BL over the Alps (Eggishorn)

Two key issues for COSMO-1 and -E:
→ ABLs over complex terrain
→ Turbulence and shallow convection in greyzone

Complex topography

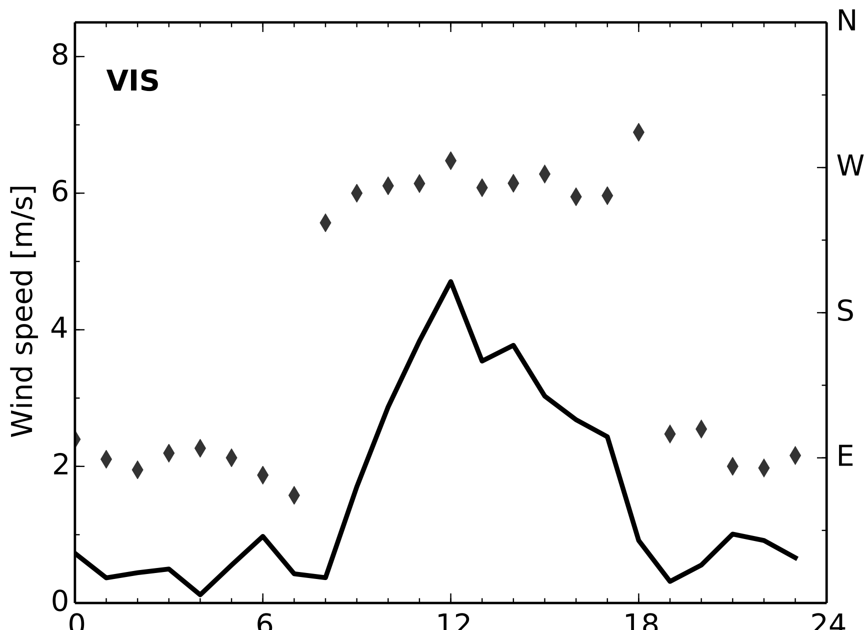
Model vs. real topography at Δx = 2.2 km (COSMO-E) at Guetsch (Andermatt)

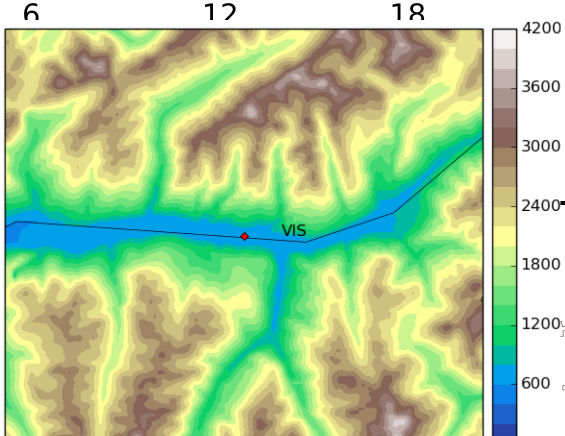


Mean diurnal cycle of valley winds

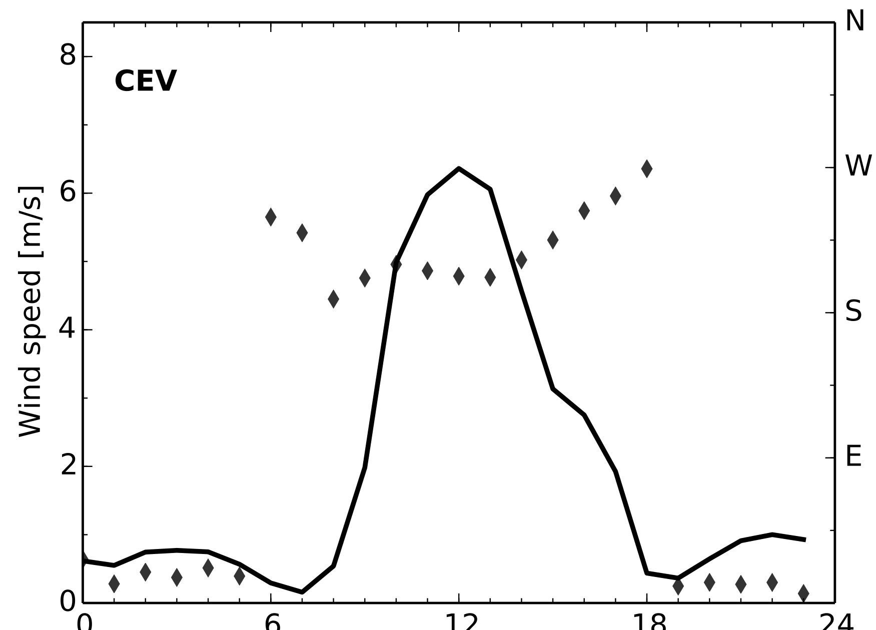
OBS

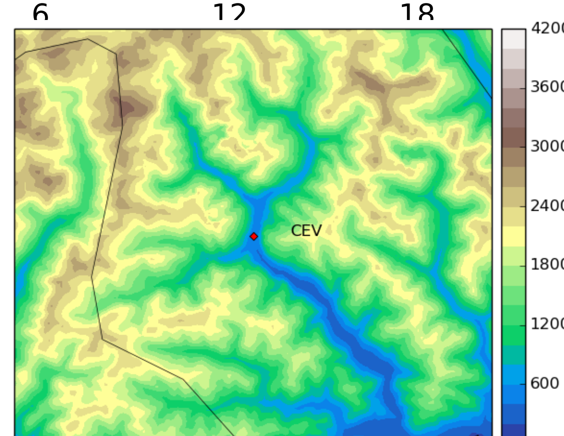
Visp (Rhone valley)





Cevio (Maggia valley)



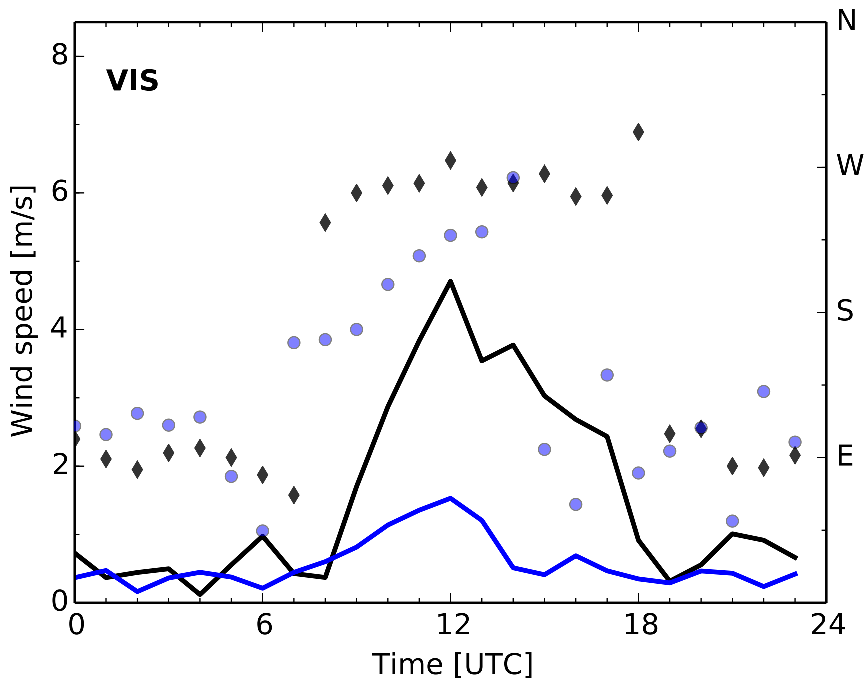


avera… –27 July)

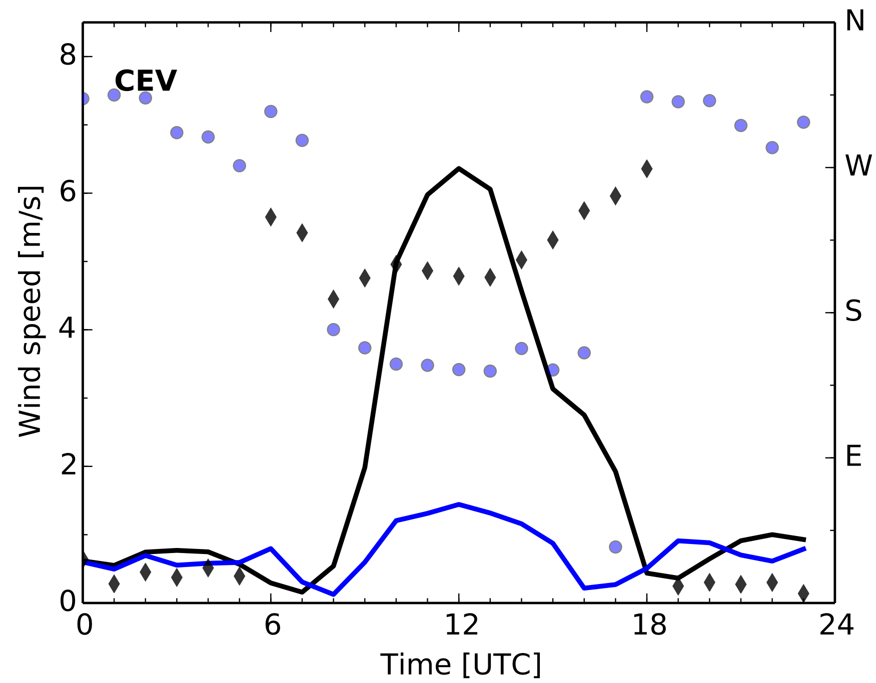
Mean diurnal cycle of valley winds

OBS 2 km

Visp (Rhone valley)



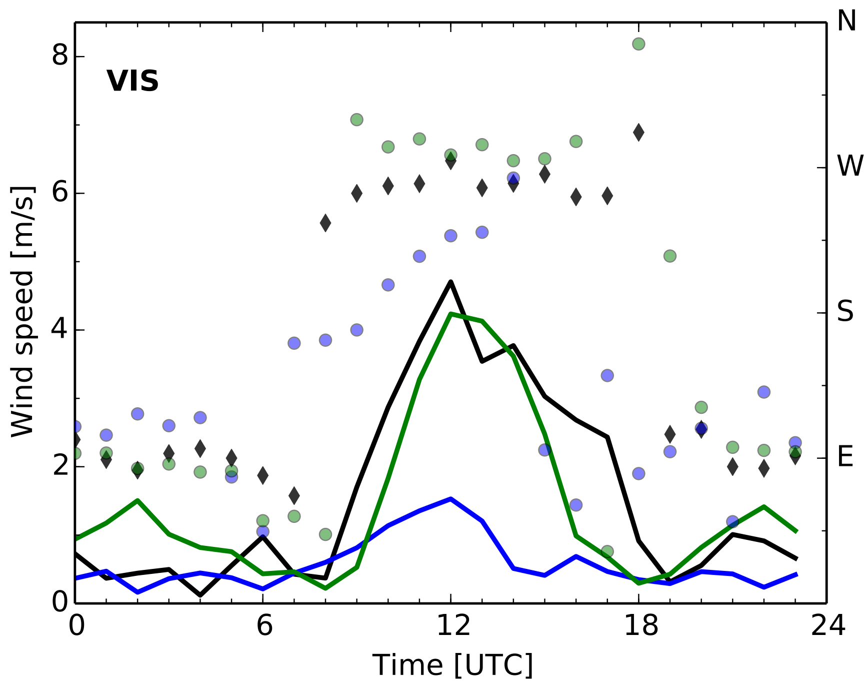
Cevio (Maggia valley)



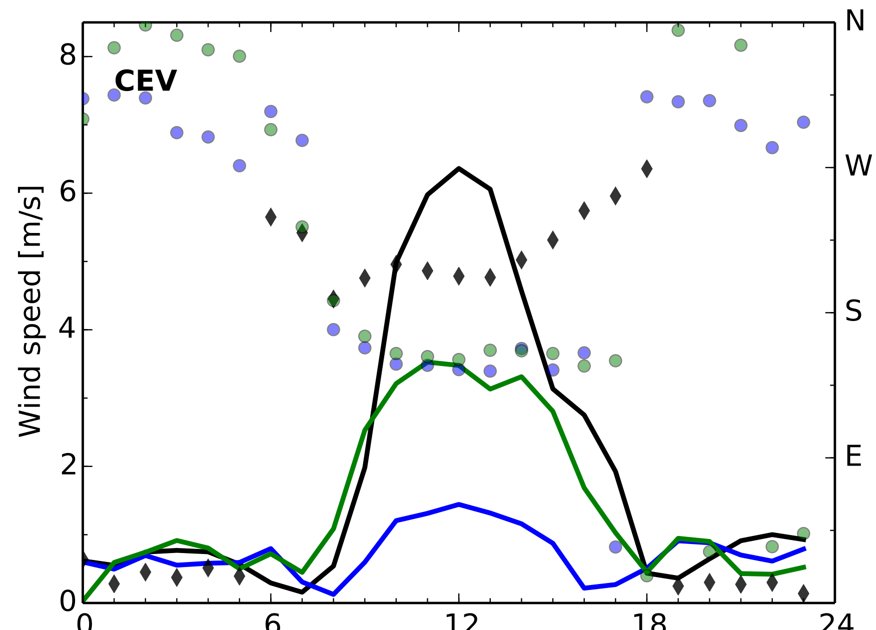
Mean diurnal cycle of valley winds

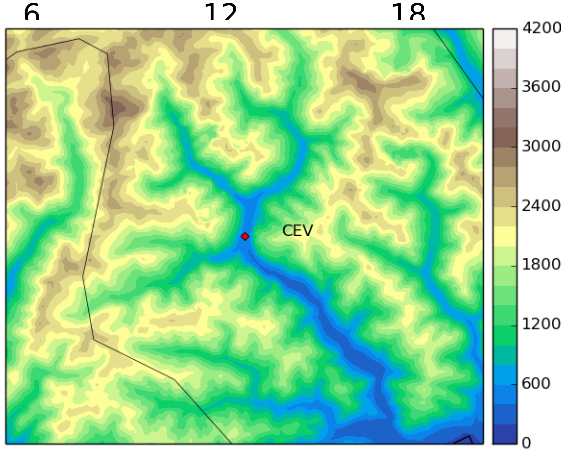
OBS 2 km 1 km

Visp (Rhone valley)

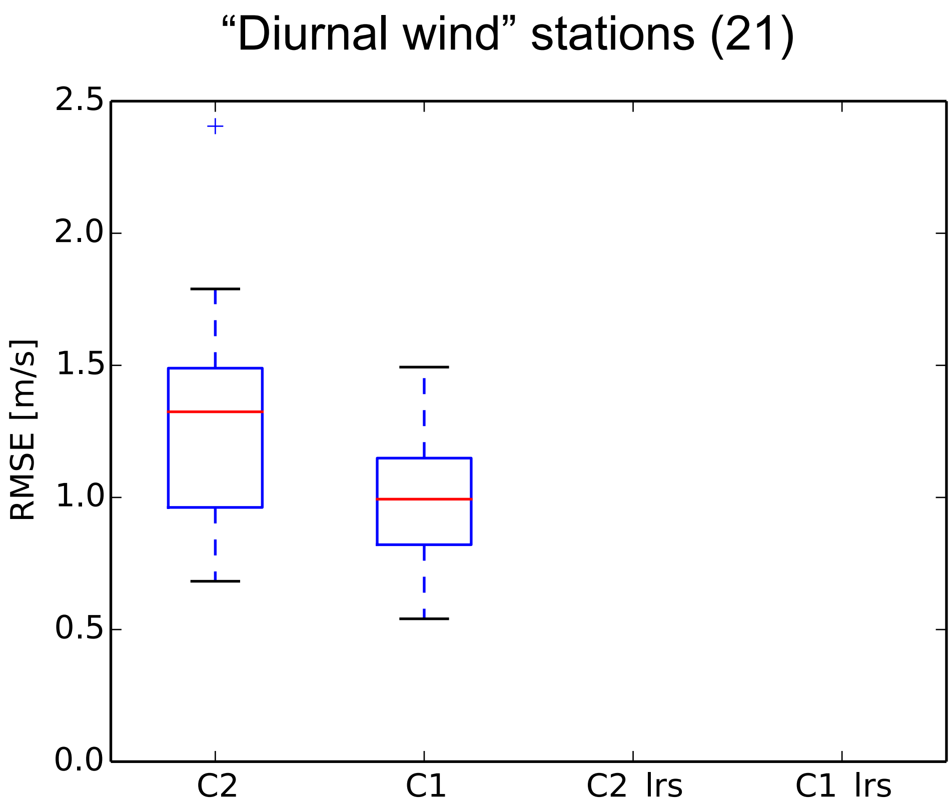


Cevio (Maggia valley)





Influence of surface data



High-resolution surface data

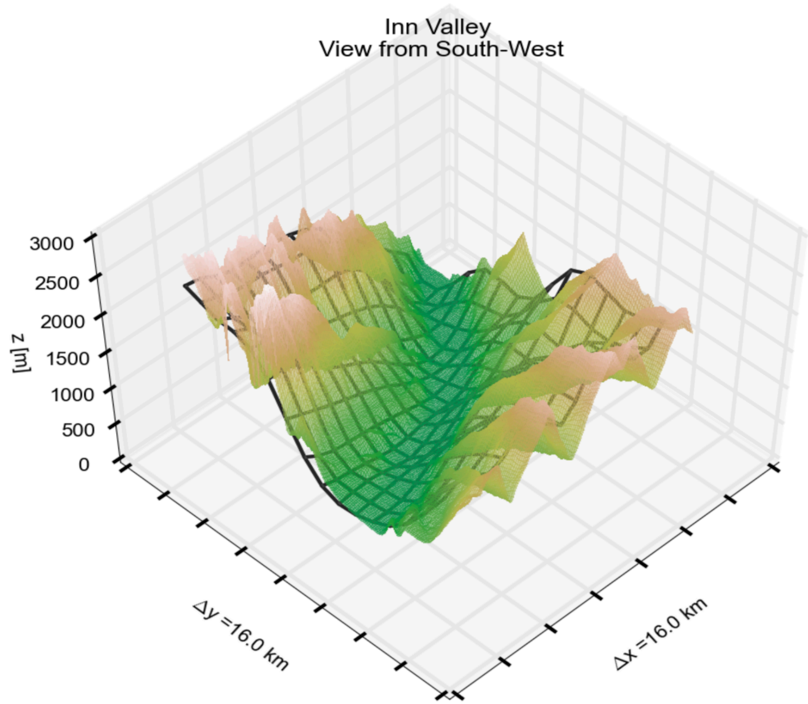
- ASTER topography (30 m)
- GC2009 land cover (300 m)
- HWSD soil type (1 km)
- Raymond filter for topography (def: cutoff ~5 dx)

Low-resolution surface data (lrs)

- GLOBE topography (1 km)
- GLC2000 land cover (1 km)
- FAO DSMW (10 km)
- Raymond filter for topography (def: cutoff ~5 dx)

→ Coarse surface data: Only minor improvement for 1km!
→ Need high-resolution surface data for 1km simulation!

TKE budget in the Inn valley (i-Box)

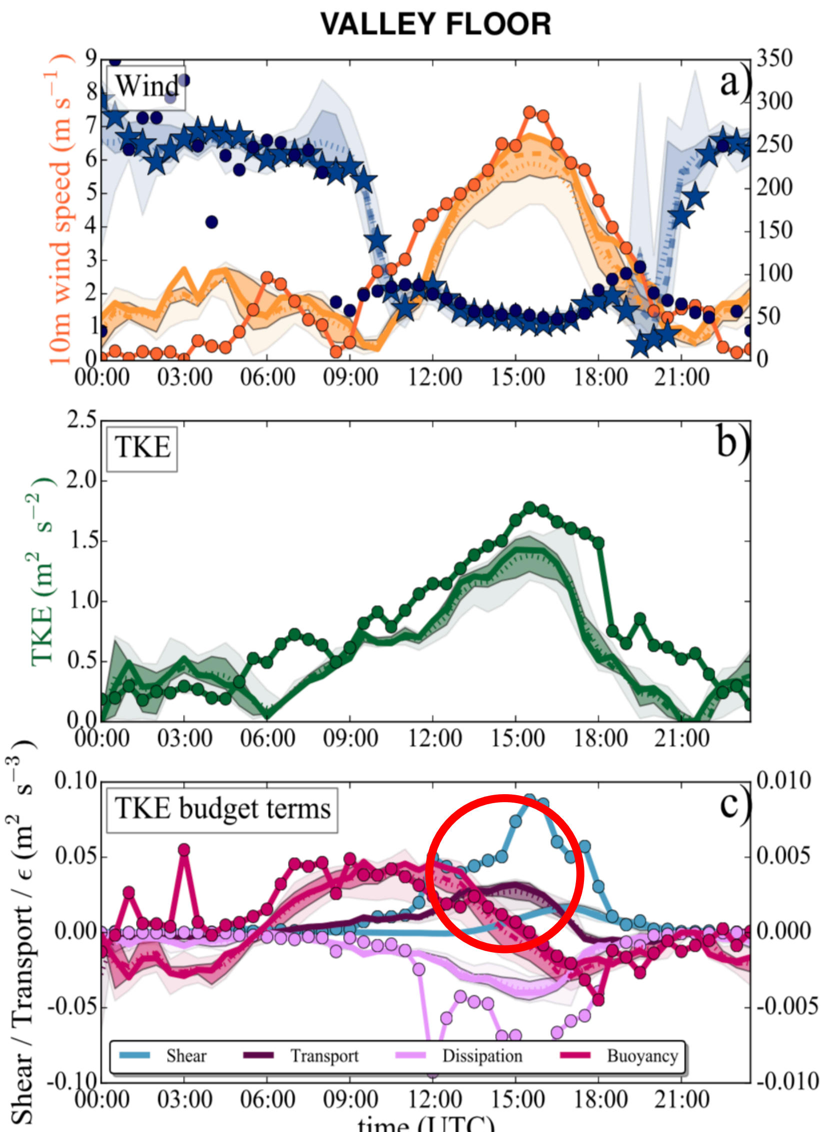


Kohlsass station (545 masl)

- Compare COSMO-1 against measurements (turbulent fluxes, TKE, TKE production terms)

1D Scheme

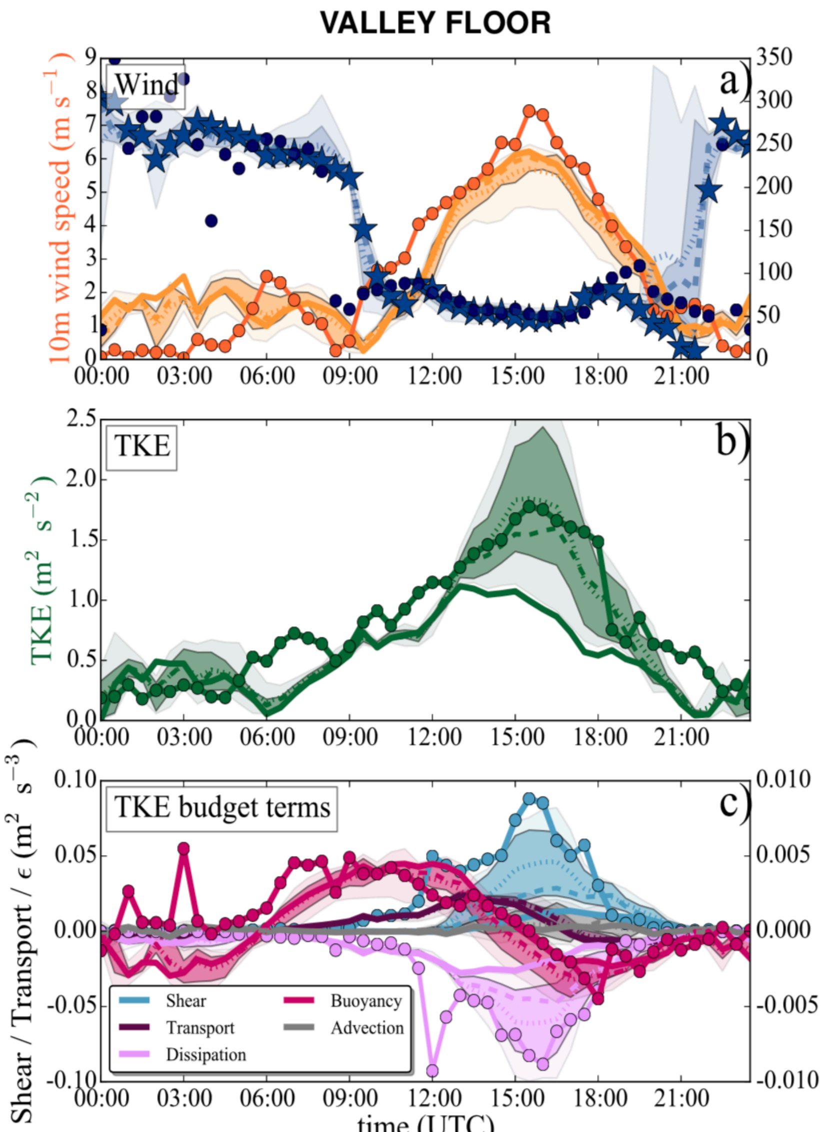
(1.5 order TKE closure)



- **Morning**
  - Buoyant production dominates
  - TKE well simulated by model
- **Afternoon**
  - Vertical shear generation by valley wind
  - **Shear term drastically underestimated** (missing horizontal contributions)

Hybrid Scheme

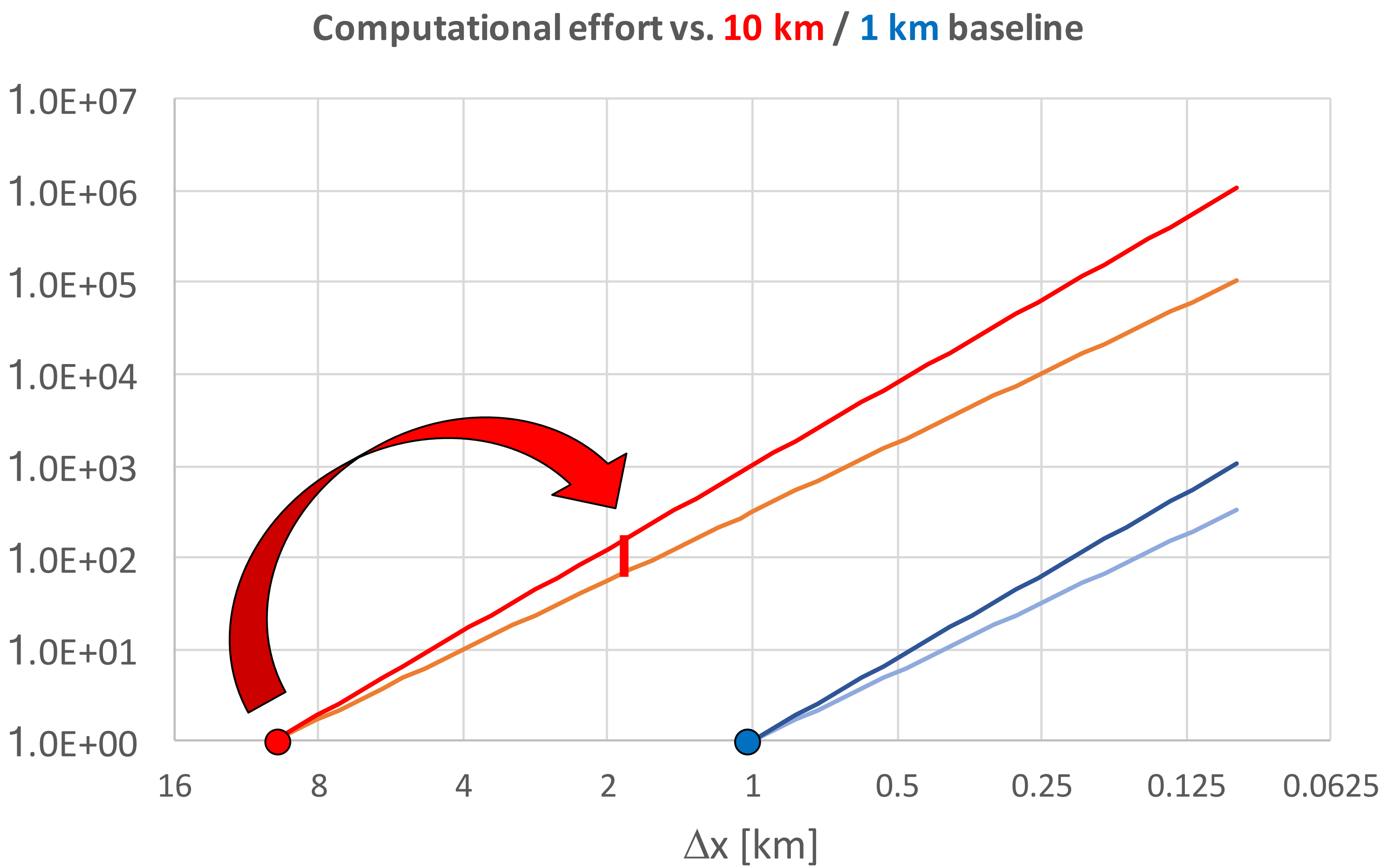
(+ TKE advection + horizontal shear contribution after Smagorinsky & Lilly)



- Significant improvement in afternoon shear production

Many shades to grey…

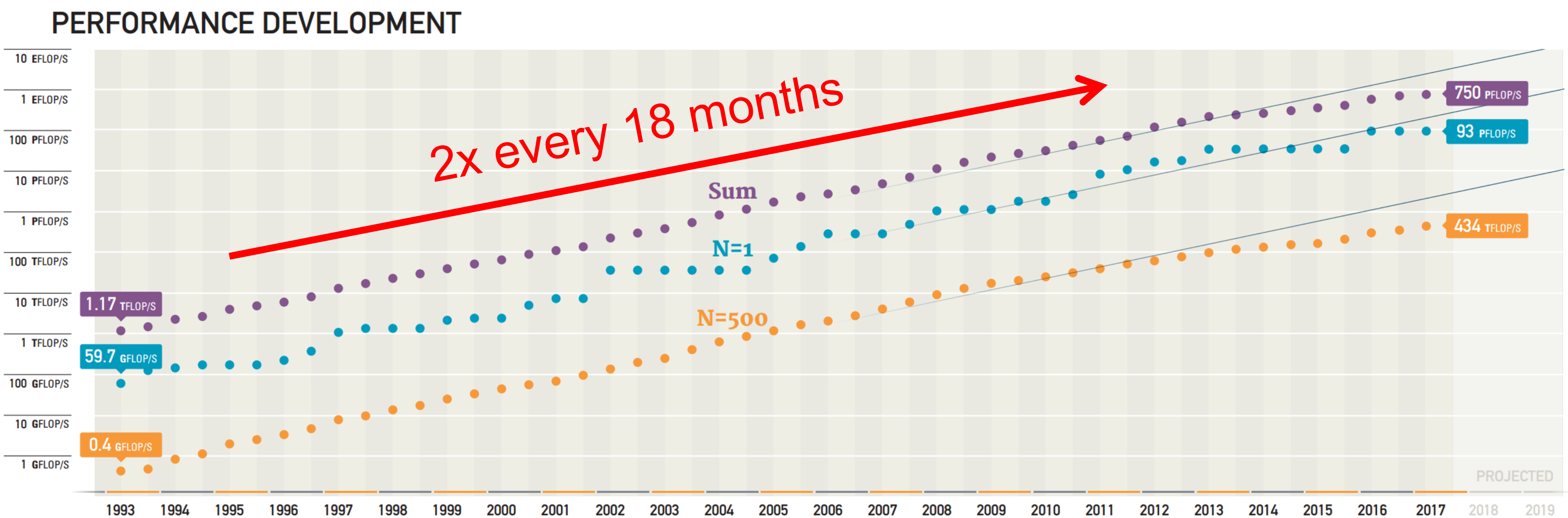
Jump across greyzone?



Computational effort grows by a factor ~100

How to achieve a factor 100?

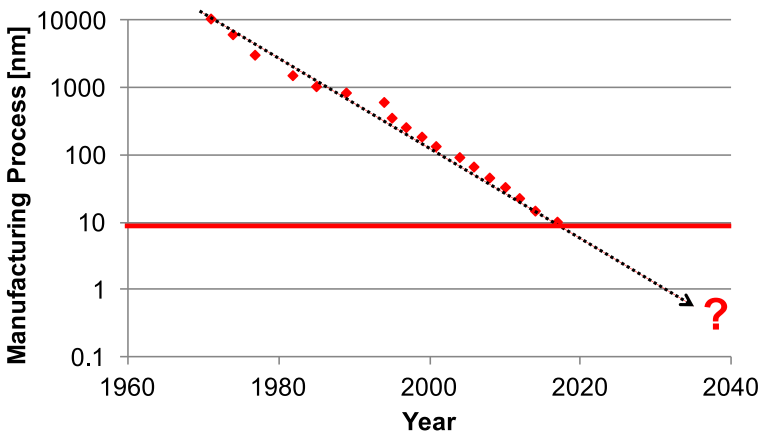
- Option 1: Money
- Option 2: Wait 10 years (Moore's law)



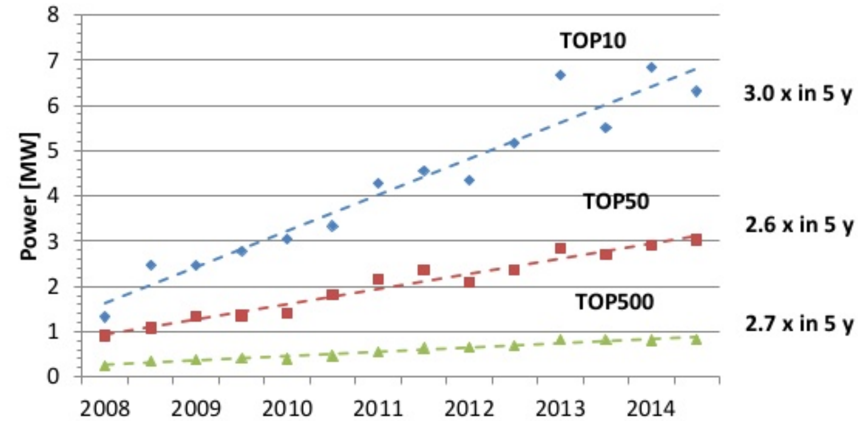
Source: top500.org

Moore's law is sick (or dead)

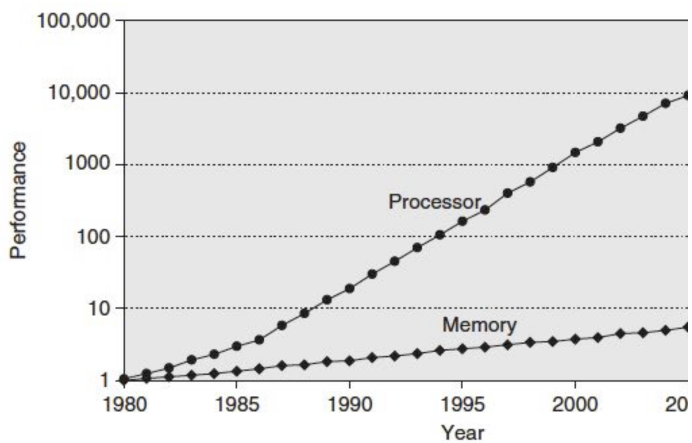
Decreasing feature size



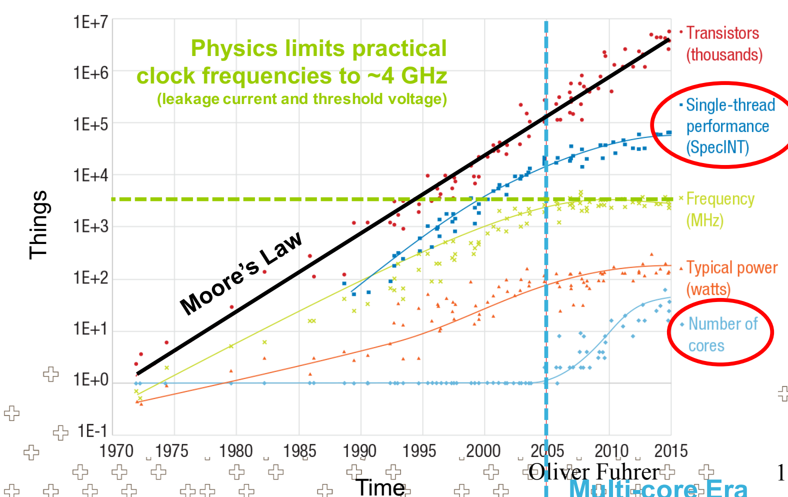
Increasing power consumption



Slower pace for bandwidth



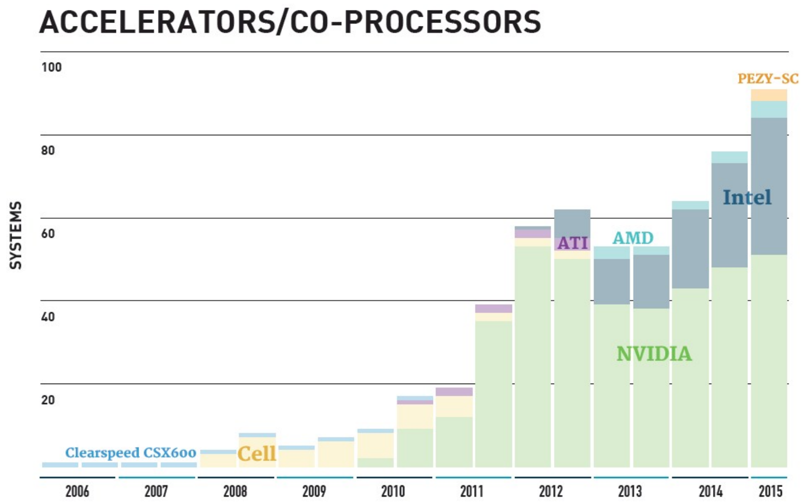
Stagnating clock frequencies

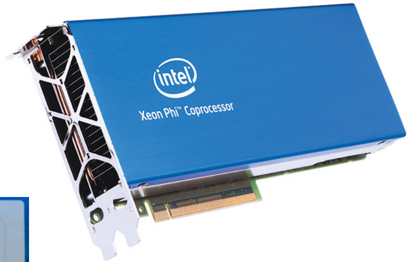


MeteoSwiss

Specialization

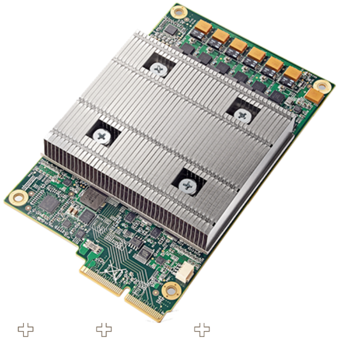
20% Systems on Top 500





Many-core processors

GPUs



AI Accelerators
(e.g. Google Tensor Flow,
Intel Nervana)

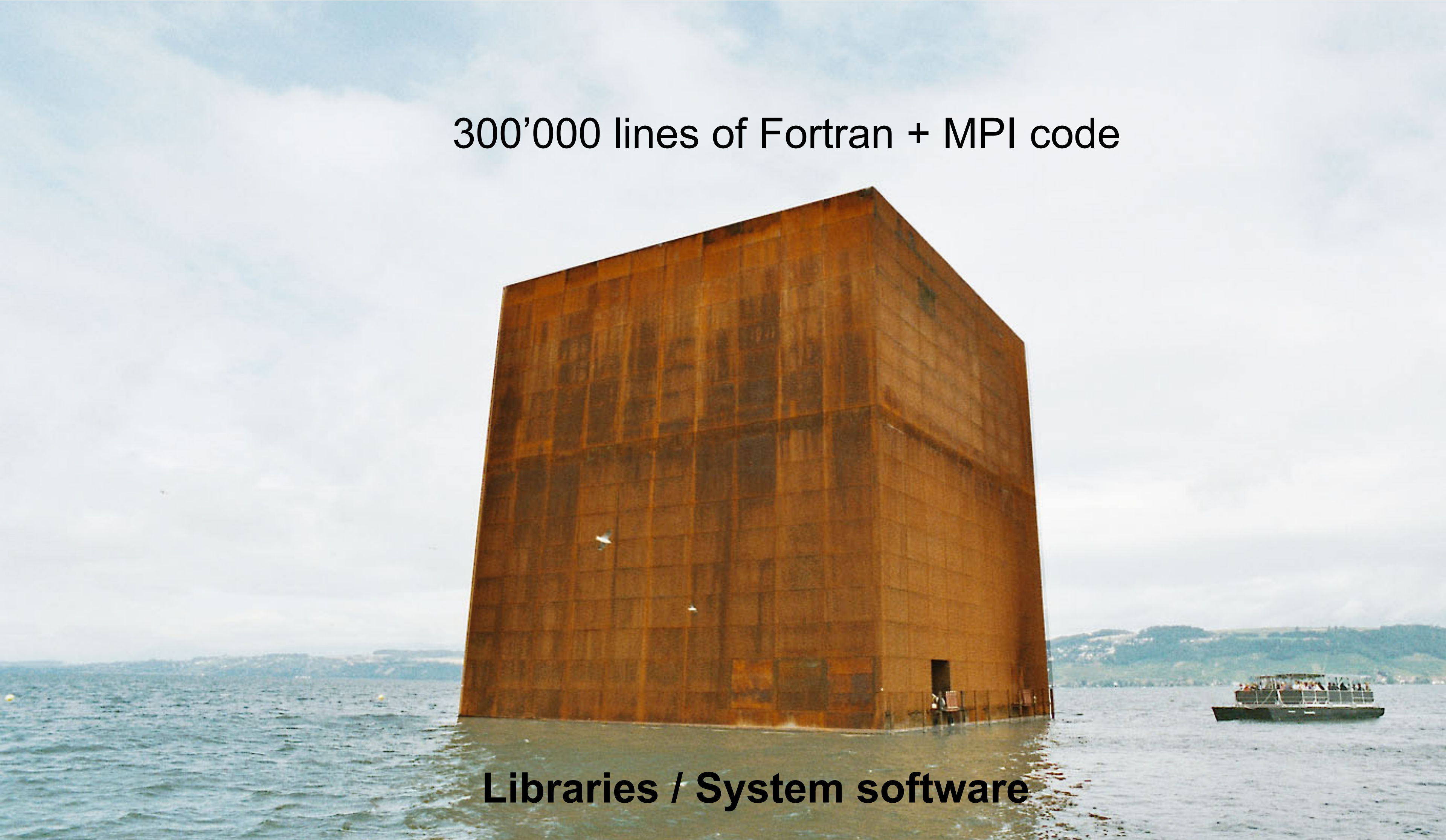
Koenig et al. 2017, UCB
(accelerator for exact dot product)

ARM
(e.g. Mont Blanc)

Challenges

- **New design constraints**
  - Maximize parallelism
  - Minimize data movement and energy consumption
  - Minimize synchronizations
- **New programming models**
  - E.g. OpenMP 4.5, Coarray Fortran, CUDA, OpenACC
- **Rapid change**
  - Timescale of HPC system vs. Model

What to do?


300’000 lines of Fortran + MPI code
Libraries / System software

Up or down?

- **Increase level of abstraction**
  - Remove details of implementation
  - Can be “disruptive”

- **Lower level of abstraction**
  - Add implementation details
  - Often „incremental“

DOWN – Decrease level of abstraction

- **Approaches**
  - Fortran + MPI + Directives (OpenMP, OpenACC)
  - Optimize code for a specific hardware
  - Custom implementations (#ifdef) or programming languages

Original Version

- Algorithm: TDMA
- Language: Fortran
- Grid: Structured
  Data layout: (i,j,k)
- Parallelization: MPI in (i,j)
- Loop order: (jki)
- Blocking: (j)
- Vectorization: (i)
- Directives: NEC
- …

```
! solve tridiag(a,b,c) * x = d

! pre-computation
...

do j = jstart, jend


  ! forward elimination
  do k = nk, 2, -1
    do i = istart, iend
!CDIR ON ADB(d)
      d(i,j,k) = ( d(i,j,k) - d(i,j,k+1) * c(i,j,k) ) * b(i,j,k)
    end do
  end do

  ! back substitution
  do k = 1, nk-1
    do i = istart, iend
!CDIR ON ADB(x)
      x(i,j,k+1) = a(i,j,k+1) * x(i,j,k) + d(i,j,k+1)
    end do
  end do

end do
```

Optimized GPU Versi

```
! solve tridiag(a,b,c) * x = d

!$ACC DATA COPYIN(a,b,c,d) COPYOUT(x)

!$ACC KERNELS LOOP, GANG(32), WORKER(8)
do i = istart, iend
do j = jstart, jend

  ! pre-computation
  ...

  ! forward elimination
  do k = nk, 2, -1
    d(i,j,k) = ( d(i,j,k) - d(i,j,k+1) * c(i,j,
  end do

  ! back substitution
  do k = 1, nk-1
    x(i,j,k+1) = a(i,j,k+1) * x(i,j,k) + d(i,j,k+1)
  end do

end do
end do
!$OMP END KERNELS LOOPS

!$ACC END DATA
```

- Algorithm: TDMA
- Language: Fortran
- Grid: Structured
- Data layout: (i,j,k)
- Parallelization: Nodes (i,j) and Blocks (i,j)
- Loop order: (ijijk)
- No Blocking
- Vectorization: SIMD Threads (i,j)
- Directives: OpenACC
- …

DOWN – Discussion

- Easy to learn
- Incremental

- Harder to understand / adapt
- Increased maintainenance effort
- Performance compromise

- **Is it possible to reach a good compromise?**
  - Near optimal performance
  - Multiple hardware architectures
  - Maintainable code

UP – Increase level of abstraction

- **Approaches**
  - Compilers
  - Libraries / Frameworks
  - Code generators and source-to-source translators
  - Domain-specific languages (DSL)

Domain-specific language (DSL)

```
function avg {
  offset off
  storage in

  avg = 0.5 * ( in(off) + in() )
}

function coriolis_force {
  storage fc, in

  coriolis_force = fc() * in()
}

operator coriolis {
  storage u_tend, u, v_tend, v, fc

  vertical_region ( k_start , k_end ) {
      u_tend += avg(j-1, coriolis_force(fc, avg(i+1, v))
      v_tend -= avg(i-1, coriolis_force(fc, avg(j+1, u))
    }
  }
}
```

Example: gtclang

- Coriolis force

$$\frac{\partial u}{\partial t} = \ldots + fv$$
$$\frac{\partial v}{\partial t} = \ldots - fu$$

- No loops
- No data structures
- No halo-updates

UP – Discussion

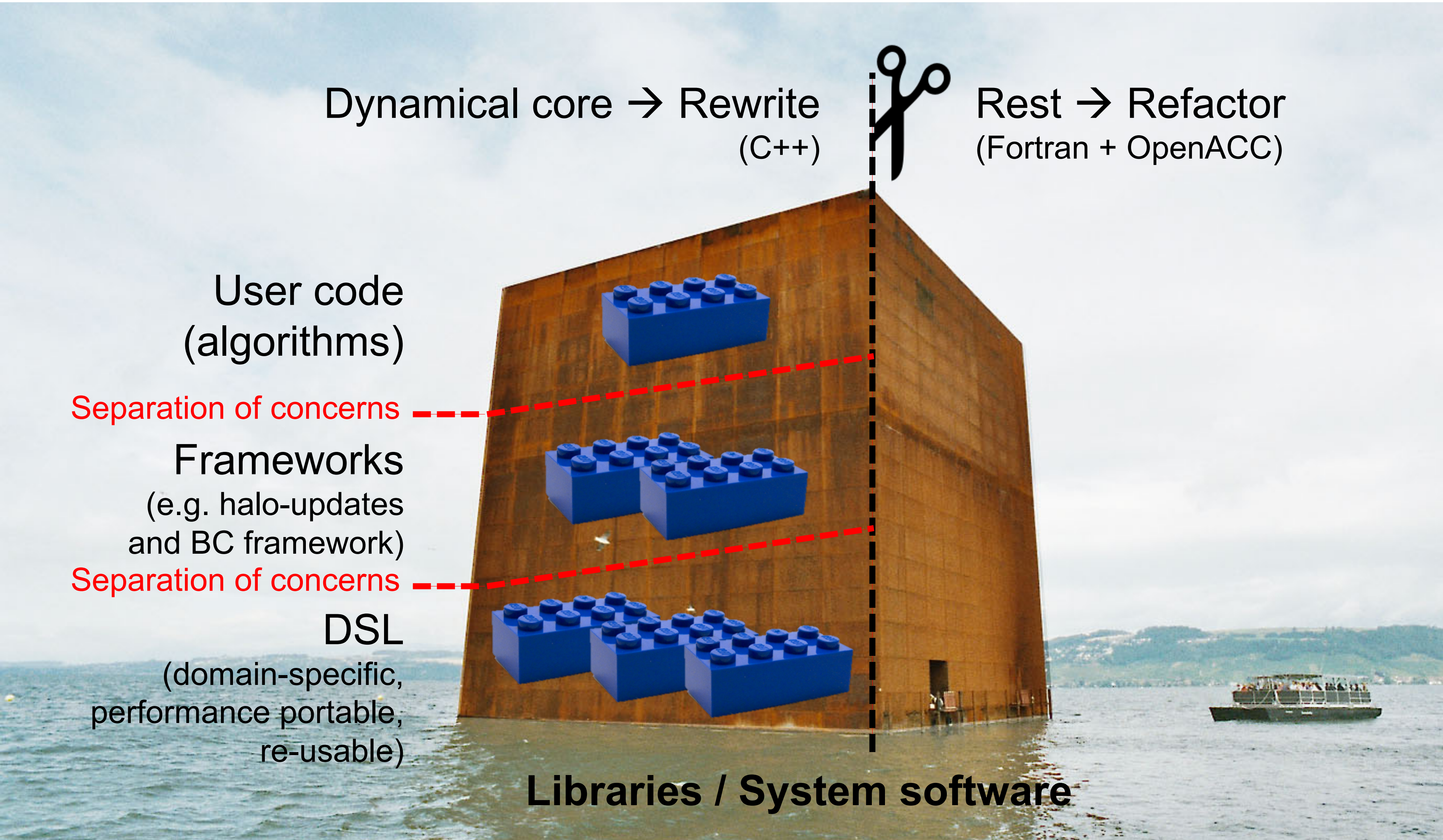
- User code easy to understand / modify
- Performance portability
- High performance
- Safety / correctness can be imposed

- No turn key solutions available
- Disruptive change
- Maintenance of DSL / compiler

- **Can be achieve a community solution?**

Both approaches for COSMO

Fuhrer et al., 2014, doi:10.14529/jsfi1401
Lapillonne and Fuhrer, 2014, doi: 10.1142/S0129626414500030



Operational system in 2016

Piz Kesch (Cray CS Storm)

- GPU-accelerated hybrid system
- "Fat" compute nodes with
  - 2 x Intel Haswell E5-2690v3
  - 8 x NVIDIA Tesla K80
- 12 nodes per rack
- Including service nodes, post-processing nodes, file system, …
- 2 redundant racks

Increase in computational problem size of 40x in 4-5 years with constant budget and running costs.

Learnings (so far)

- **Co-design approach**: Team of computer scientists, computational scientists, domain scientists, system architects and vendors
- Trading off options and **taking compromises** is important
  - e.g. cannot re-write everything in one go
  - e.g. data-assimilation is not a good match for GPUs
- Consider **full workflow** (including I/O, pre-/post-processing, …)
- Aim for **sustainable solutions**
  - Consider development and maintenance effort

How was factor 40x achieved?

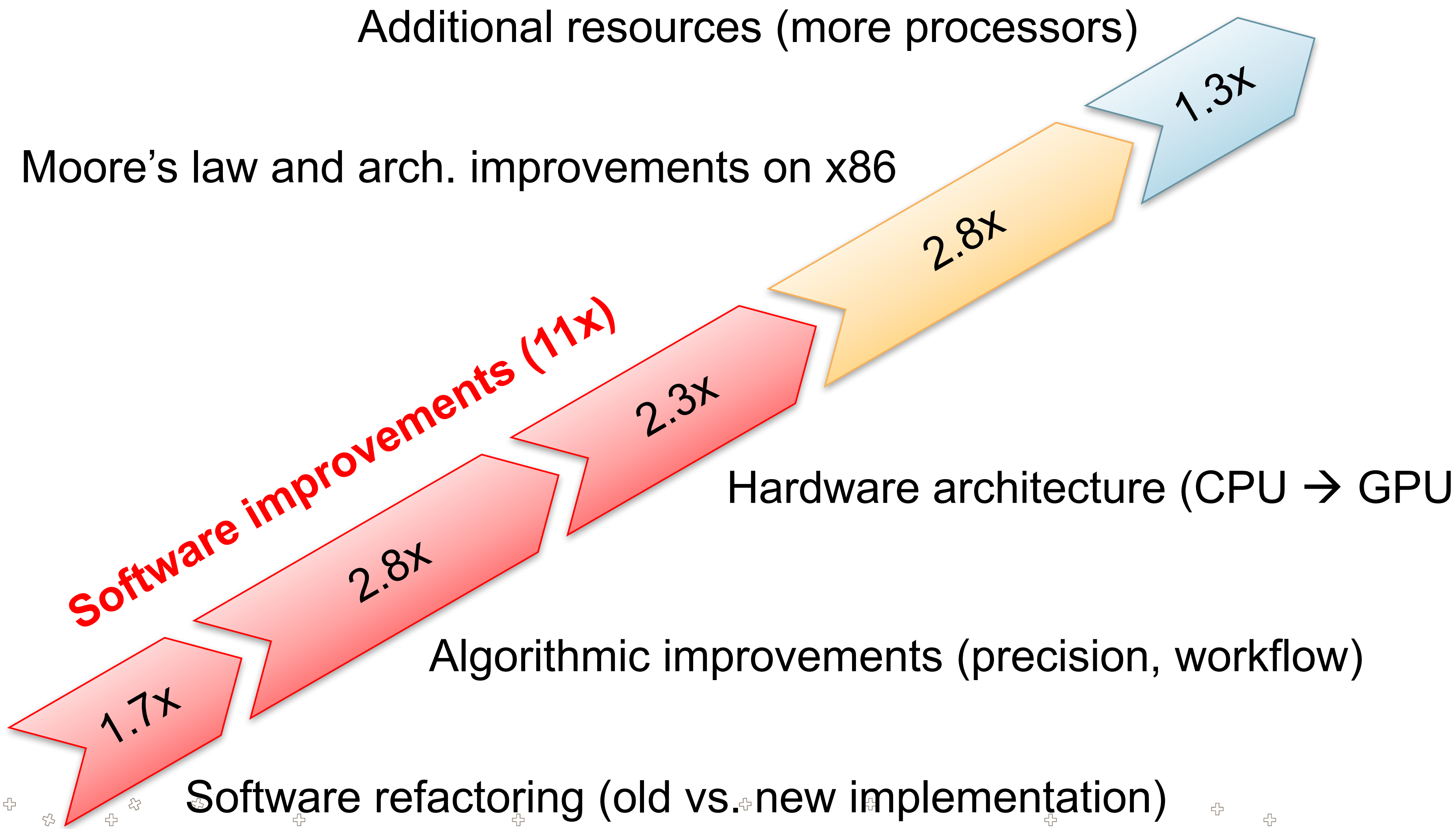
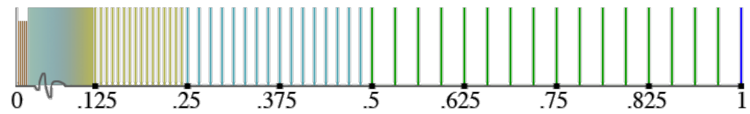


Image: http://ridiculousfish.com/blog/posts/float.html

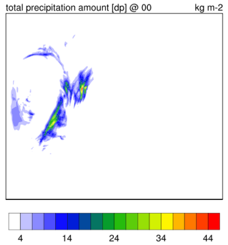
Greyzone of precision

- Discretization of number space

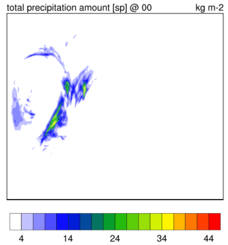


- Higher precision → more data movement / energy

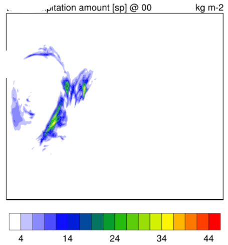
"resolved"



"work on parameterizations and algorithms"



"detrimental"



bits

64

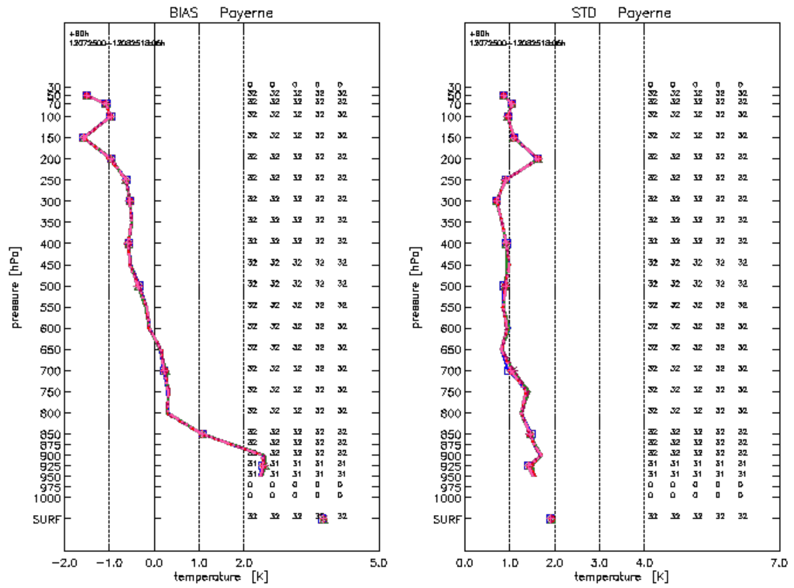
32
(1.7x faster)

16

- Tradeoff: algorithm, precision, resolution, ensemble size

COSMO in single precision

- Results not distinguishable between single / double precision (e.g. upper air verification)



- Ensemble runs are in single precision (60% more members)
- Issues with data assimilation code

Summary

- Key issues for O(1-2 km) modeling over complex terrain
  - ABLs over complex terrain
  - Turbulence and shallow convection in greyzone
- Exponentially increasing compute power is no longer a given
  - Specialization of hardware
  - New programming models
- Urgent need to also address HPC issues in order not to get stuck in greyzone!

MeteoSwiss
Operation Center 1
CH-8058 Zurich-Airport
T +41 58 460 91 11
www.meteoswiss.ch

MeteoSvizzera
Via ai Monti 146
CH-6605 Locarno-Monti
T +41 58 460 92 22
www.meteosvizzera.ch

MétéoSuisse
7bis, av. de la Paix
CH-1211 Genève 2
T +41 58 460 98 88
www.meteosuisse.ch

MétéoSuisse
Chemin de l'Aérologie
CH-1530 Payerne
T +41 58 460 94 44
www.meteosuisse.ch