Progress report on ECMWF’s Scalability Programme

Peter Bauer and the Scalability Team (RDxFDxCD, ECMWF)

Governance: ECMWF, Member states, Regional consortia

ECMWF Scalability Programme 1.0

Projects:

Observation processing:
- Lean workflow in critical path
- Object b
- Screenin

Data assimilation:
- Flexible algorithms (C++)
- IFS integration
- Coupling with ocean and sea-ice

Numerical methods:
- Numerical methods
- h/v/t-discretization, multiple grids
- Prognostic variables

Model output processing:
- Broker-worker workflow
- Near-memory processing

In the <u>shorter term</u>, implement low-hanging-fruit efficiency gains in <u>present system</u> to:
- Counterbalance cost of imminent science upgrades
- Trial portability/efficiency of *present* methodologies to *existing* hardware options
- Support planning (procurements w/ realistic budget requests, benchmarks, etc.)

In the <u>longer term</u>, test prepare and assess not-so-low-hanging fruit-efficiency gains in <u>future system</u> to:
- Counterbalance cost of more forward-looking science upgrade options
- Trial portability/efficiency of *future* methodologies to *future* hardware options
- Support planning (procurements w/ realistic budget requests, benchmarks, etc.)

ECMWF

The implications of fulfilling short-term and long-term needs are entirely different!

Weather & climate computing and data roadmap in H2020

| 2018 | 2019 | 2020 | 2021 | 2022 | 2023 | 2024 |

FETHPC/ICT:

1. Weather and climate benchmarks, and IO (HPCW)
2. Demonstration of novel programming models (DSL)
3. Data aware numerical methods

→ Feasibility of new concepts, computability

ESCAPE, nextgenio, EuroEXA, ESCAPE 2, MAESTRO DATA ORCHESTRATION, EPiGRAM HS, LEXIS

EINFRA/ INFRAEDI:

1. Full-sized weather and climate models for EuroHPC
2. Community testing of novel programming models (DSL)
3. Data handling workflows, data analytics research

→ Adaptation of leading models to (pre-)exascale; strategy for achieving full-sized requirements

esiwace, HiDALGO, esiwace2 CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE

FETFLAG:

1. Full-sized applications with required speed/volume and power footprint
2. Ingestion of downstream applications, all ensembles
3. Domain-specific, distributed computing capability, interactive workflows

→ Redesign entire prediction philosophy

ExtremeEarth

EuroHPC JU:

EuroHPC-1&2 pre-exascale

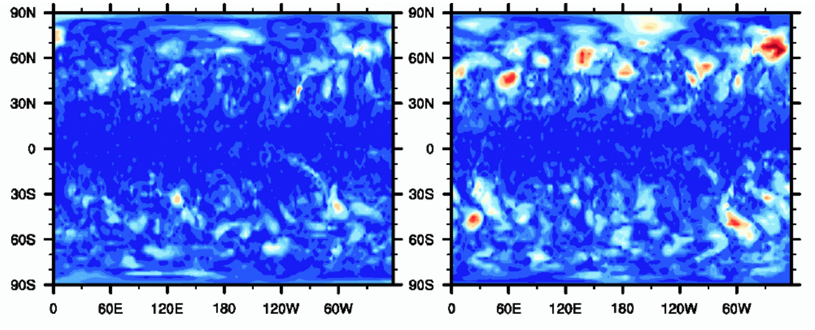
→ New centralized European infrastructure

EuroHPC-3&4 exascale

Low(ish)-hanging fruit: Computing

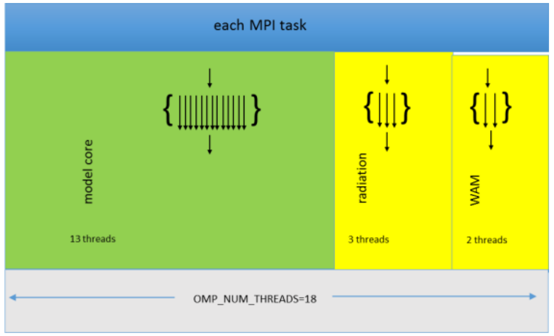
Funded by the European Union

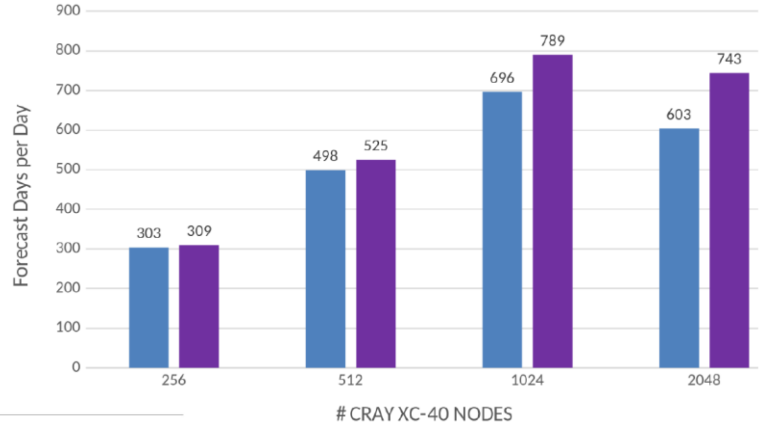
Day-10 forecast difference SP vs DP (T in K at 850 hPa) | Day-10 ensemble spread all DP (T in K at 850 hPa)



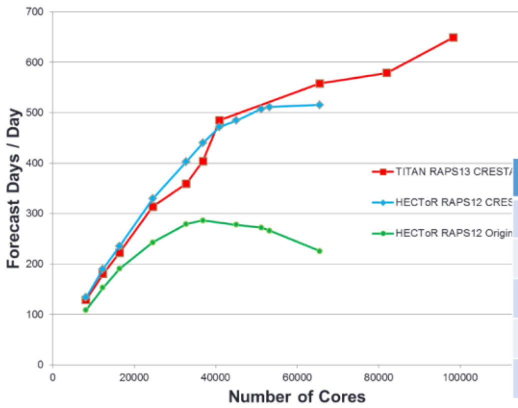
Precision:
- running IFS with single precision arithmetics can save 40% of runtime, IFS-ST offers options like precision by wavenumber, only for LT, in semi-implicit solver;
- storing ensemble model output at reduced precision can save 67% of data volume;

Concurrency:
- allocating threads/task (/across tasks) to model components like radiation or waves can save 20% (gain increases with resolution);
- implementation is cumbersome;





Overlapping communication & computing:
- through programming models (Fortran co-array vs GPI2 vs MPI), gave substantial gains on Titan w/Gemini,
- on XC-30/40 w/ Aries there is no overall performance benefit over default MPI implementation;

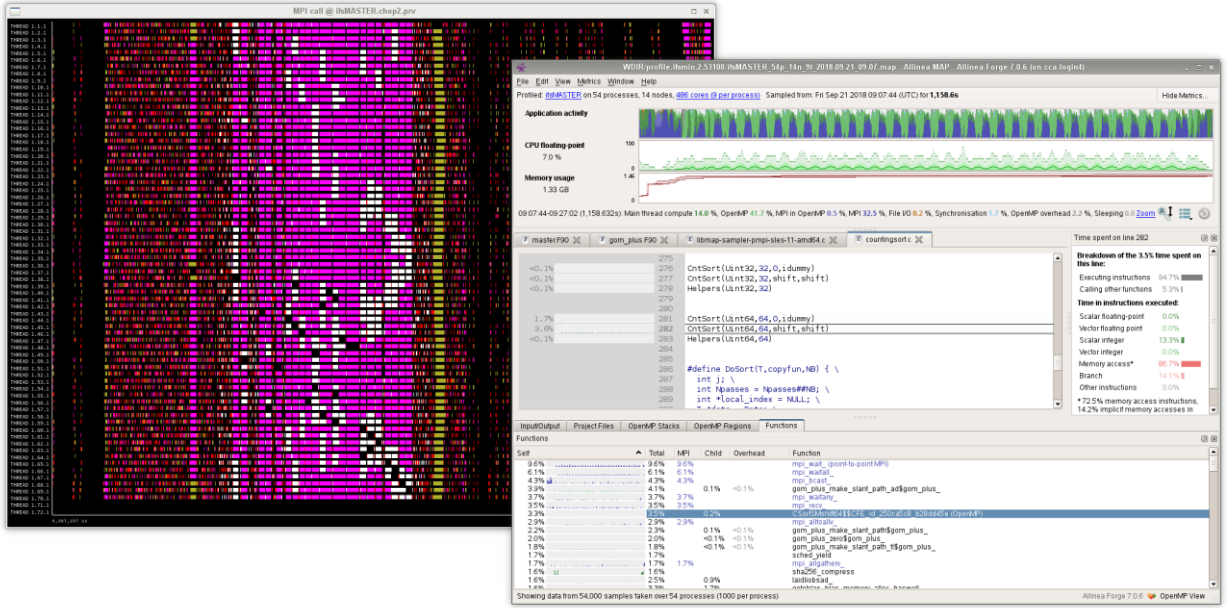


Tasks x threads	Nodes	Experiments	Forecast days/day
2160Tx12t	360	control	**1104.9**
2160Tx12t	360	control	1116.1
2160Tx12t	360	coarray2	815.6
2160Tx12t	360	coarray2	846.0
2160Tx12t	360	gpi2	788.9

ECMWF EUROPEAN CENTRE FOR MEDIUM-RANGE WEATHER FORECASTS

[Düben et al., Vana et al., Mozdzynski et al.]

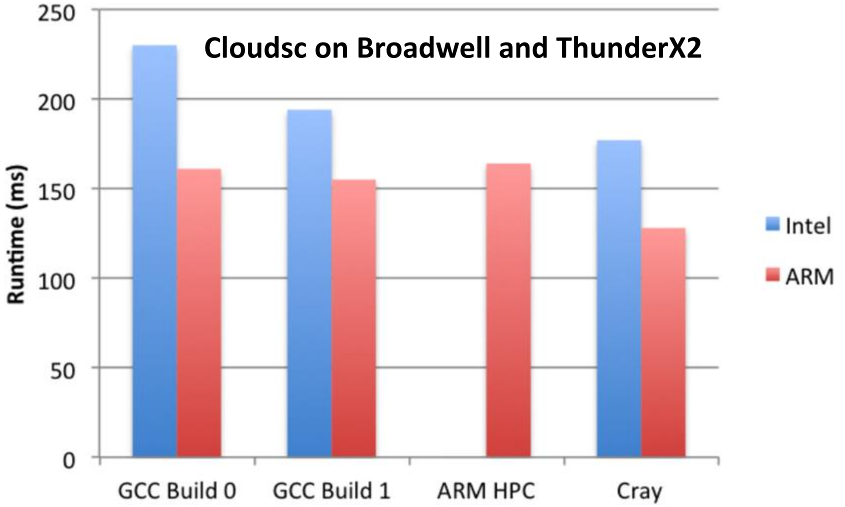
Low(ish)-hanging fruit: Diagnostics & Architectures



Performance tools:
- Integrate easy-to-use performance tools with IFS, available to all
- ARM Forge MAP, BSC Extrae & Paraver (see POP CoE)

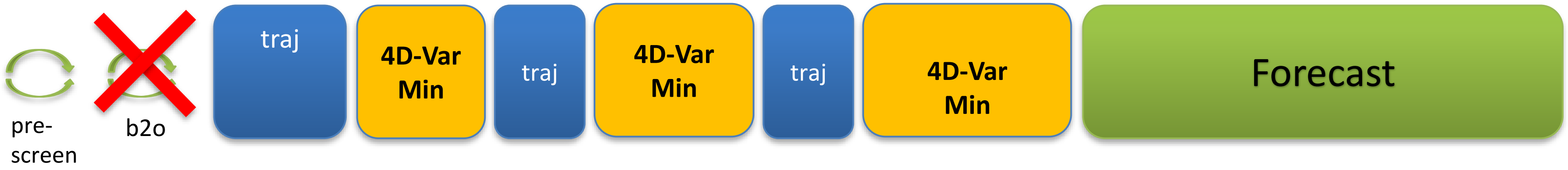
[From Patrick Gillies – check also Mario Acosta's talk on Wednesday!]

Porting code to other processor types:
- OpenIFS and ESCAPE dwarfs ported to early access nodes - collboration with U Bristol using Isambard Cray platform with Cavium ThunderX2 CPUs
- Long and short-wave MCICA solvers ported to GPU V100 with OpenACC (achieves 85% of peak memory bandwidth on V100) – collaboration with NVIDIA by hackathon for ECMWF staff

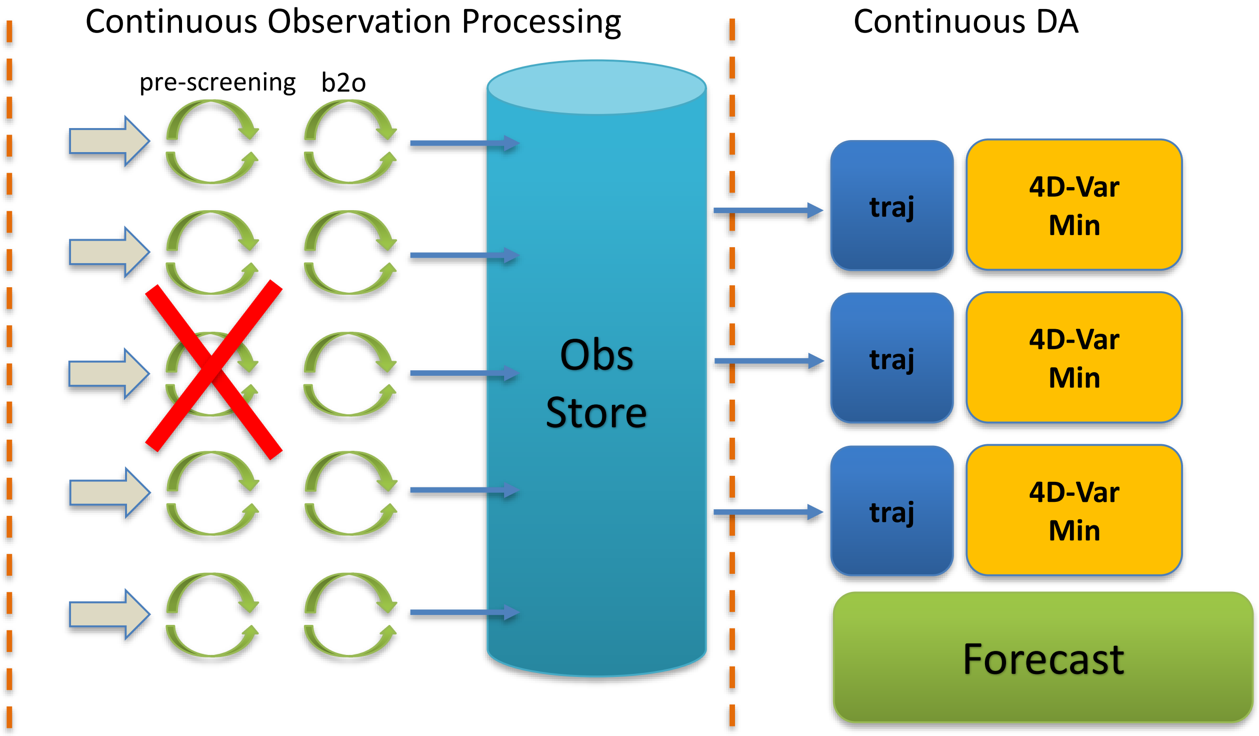


Not-so-low-hanging fruit: Pre-processing

Current processing chain is sequential; a failure at any point leads to delay in forecast production



COPE: Observations pre-screened in small batches as they arrive.
Decoupled system is more robust to failures.



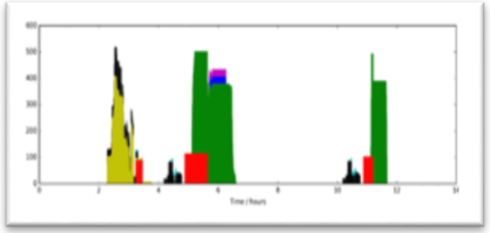
Gains:
- resilience
- 15% cost in critical path

[From Peter Lean – check his talk on Friday!]

Not-so-low-hanging fruit: Benchmarking

Funded by the European Union

- **Kronos tests HPC systems by deploying realistic workloads:**
  1. a workload model is generated from **HPC workload profiling data**
  2. the workload model is then translated (and scaled) into a **schedule of representative and easily-portable applications**
  3. Kronos models and tests **Compute, Interconnect, I/O subsystems**



HPC Workload profiles

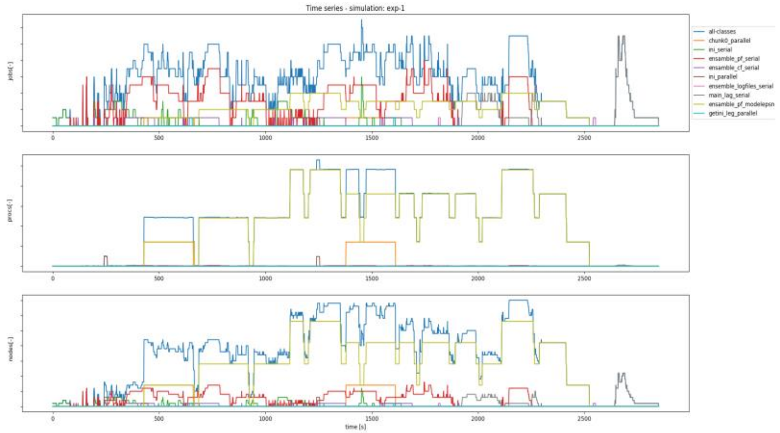
Workload Model

Synthetic Applications

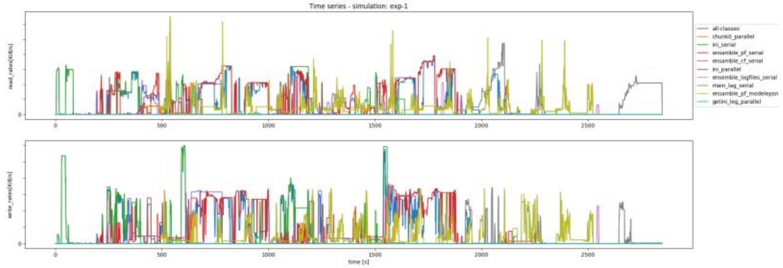
Kronos

HPC prototype to be tested

Post-processing



E.g. Workload execution profiles

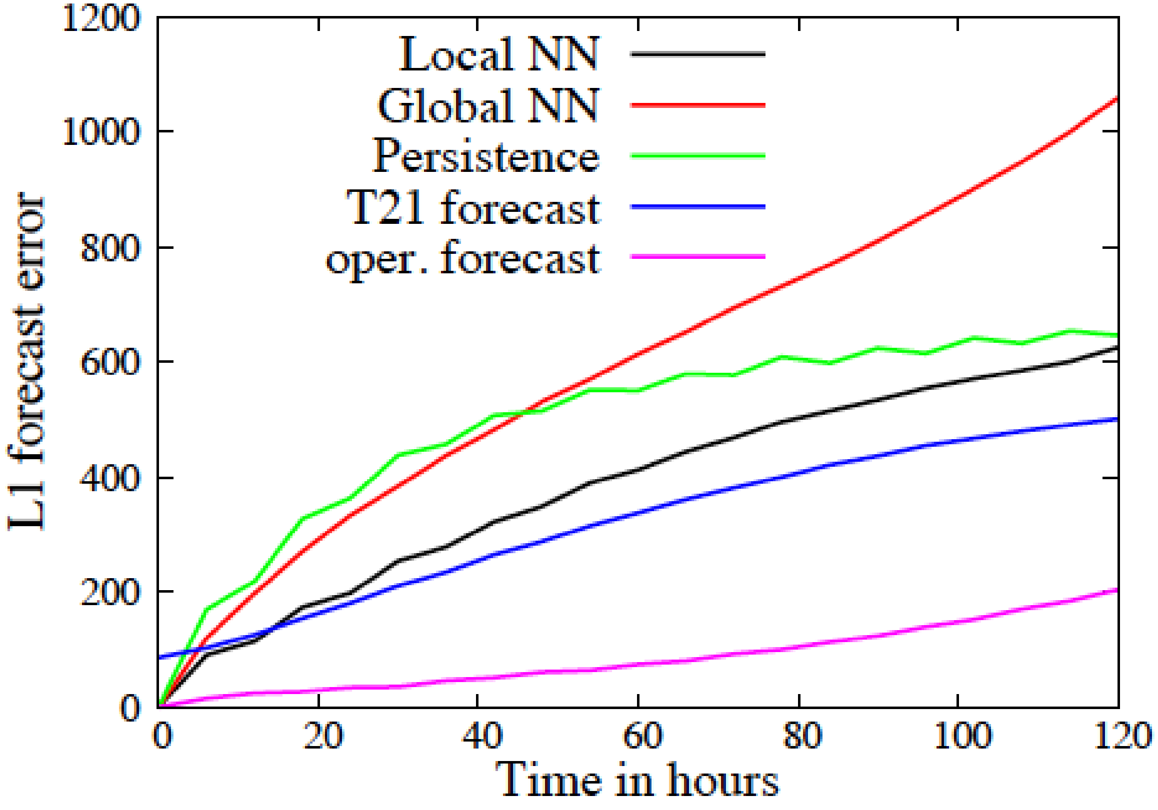


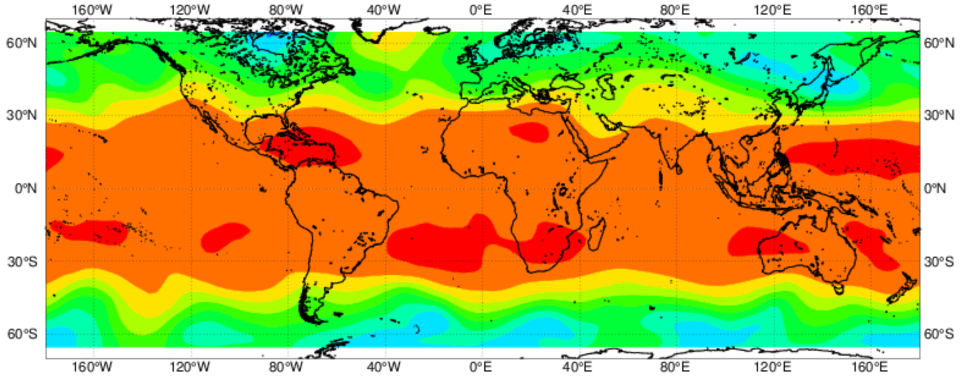
E.g. I/O time-profiles

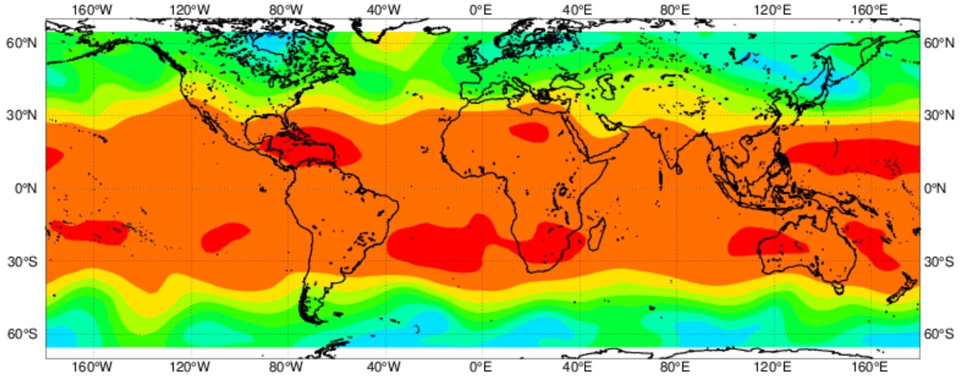
[Antonino Bonanni, Tiago Quintino]

Not-so-low-hanging fruit: AI methods for forecasting

1. Take ERA-5 z500/6° LAT/LON reanalyses/forecasts forecasts = operational forecasts, T21 forecasts, persistence
2. Train NN with truth
3. Run NN forecasts for z500 with all 9x9 grid points predicting tendency = local NN
4. Run NN forecasts for z500 with all grid points predicting tendency = global NN

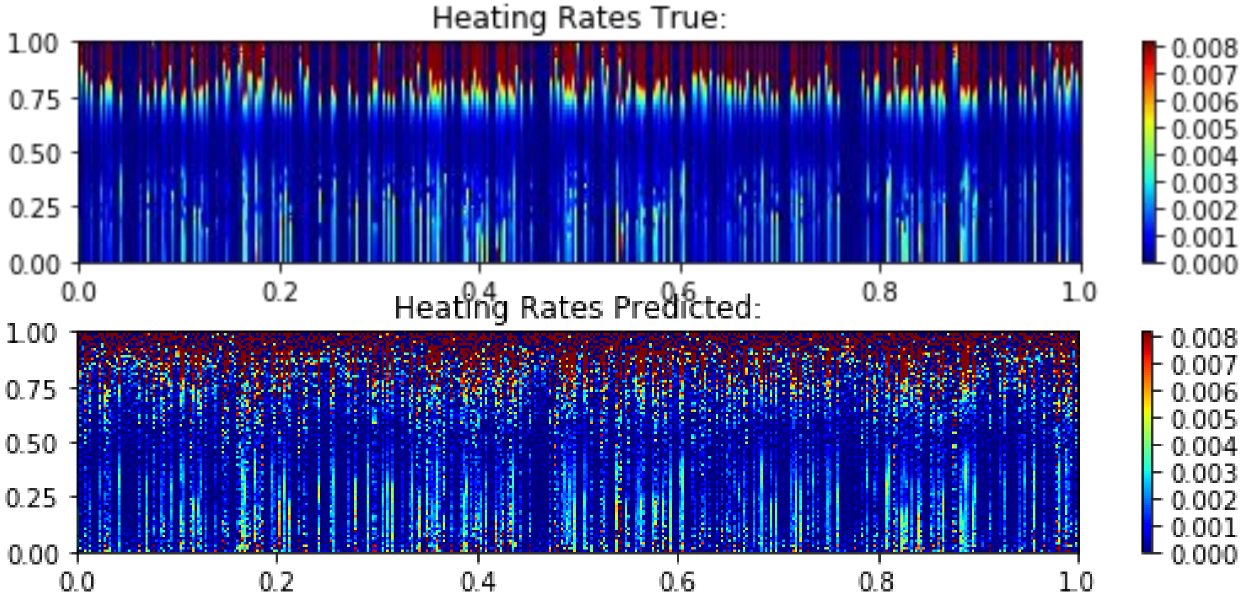






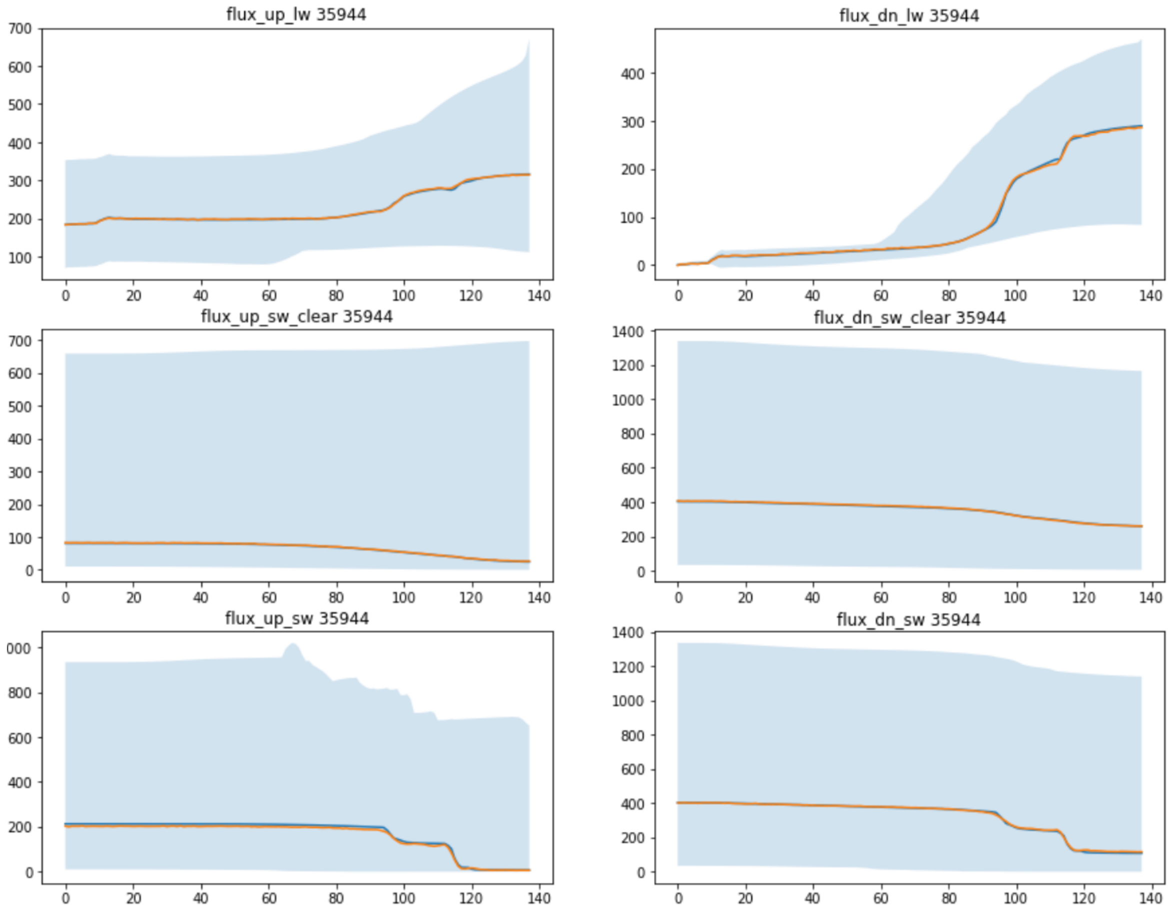
[Düben and Bauer 2018]

Not-so-low-hanging fruit: AI methods for parameterizations



Shortwave and longwave flux profiles

(reference, NN, shading = natural variability)



Data Set: 150,000 profiles total (25,000 locations with different solar zenith angles), divided into training=126,000, validation=24,000

Input to the network: 128 x 137 x 19 (128 batch size, 137 full levels, 19 variables SW clear sky)

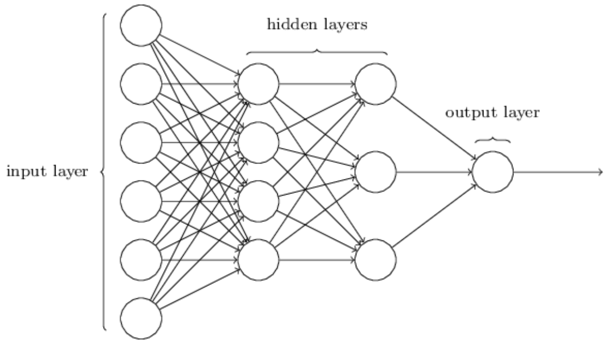
Output of the network: 128 x 138 x 2 (up and down flux on each half level)

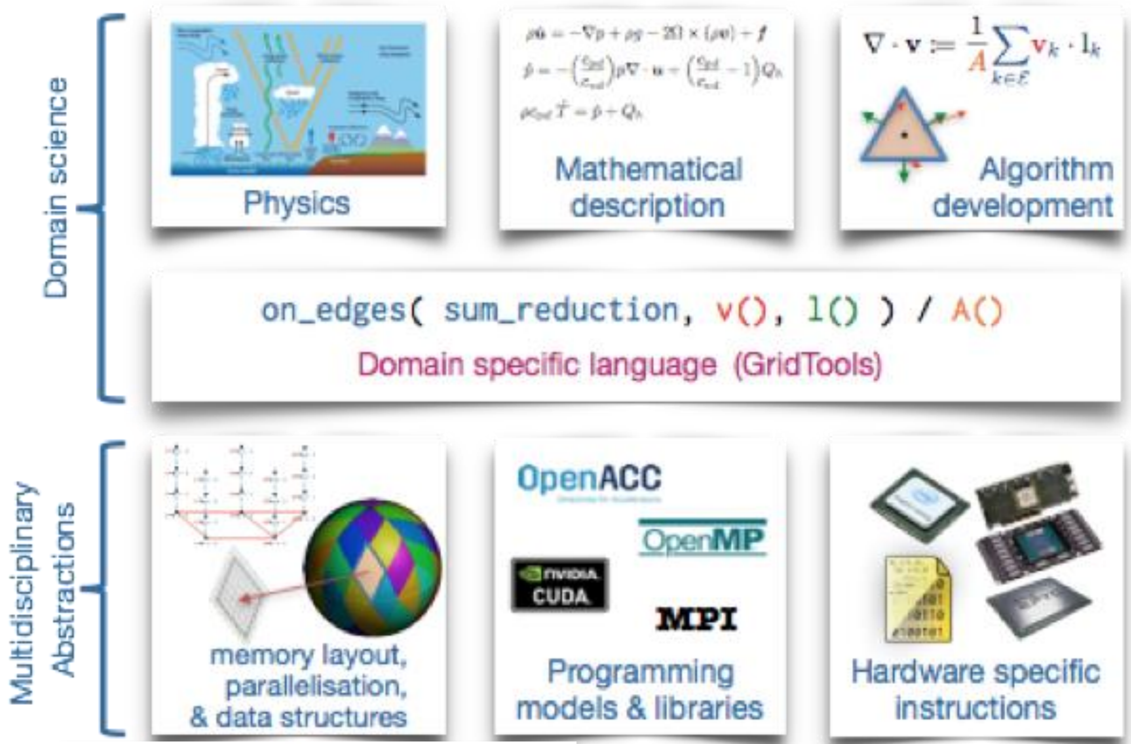
Network: four 1D convolutional layers followed by two fully connected layers; 194k trainable parameters

[Christoph Angerer & Jakob Progsch, NVIDIA; Peter Düben, Robin Hogan, Peter Bauer]

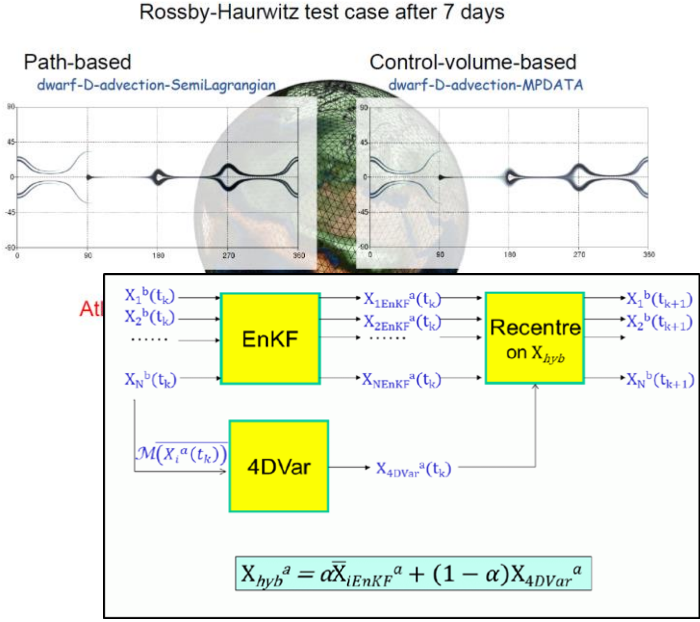
Far-hanging fruits: Algorithms – programming - hardware

Neural networks

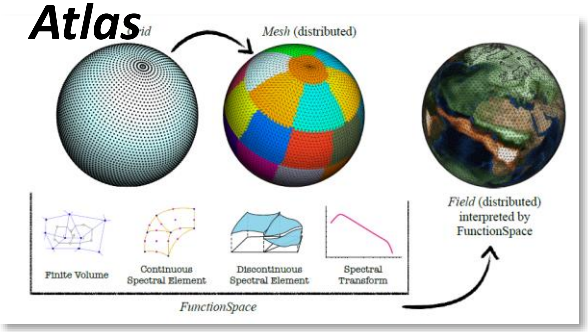




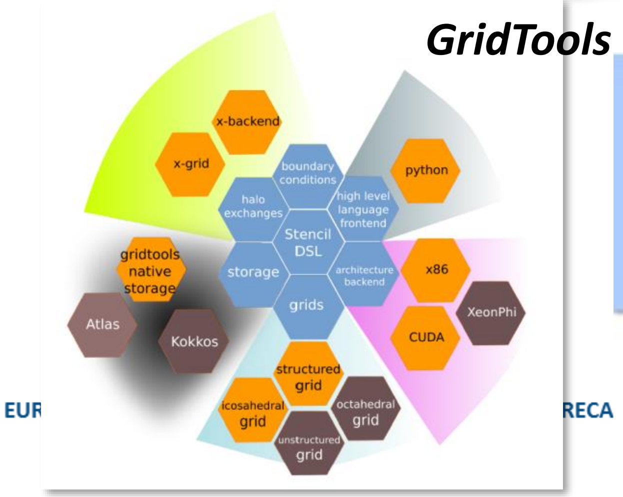
Mathematics&algorithms



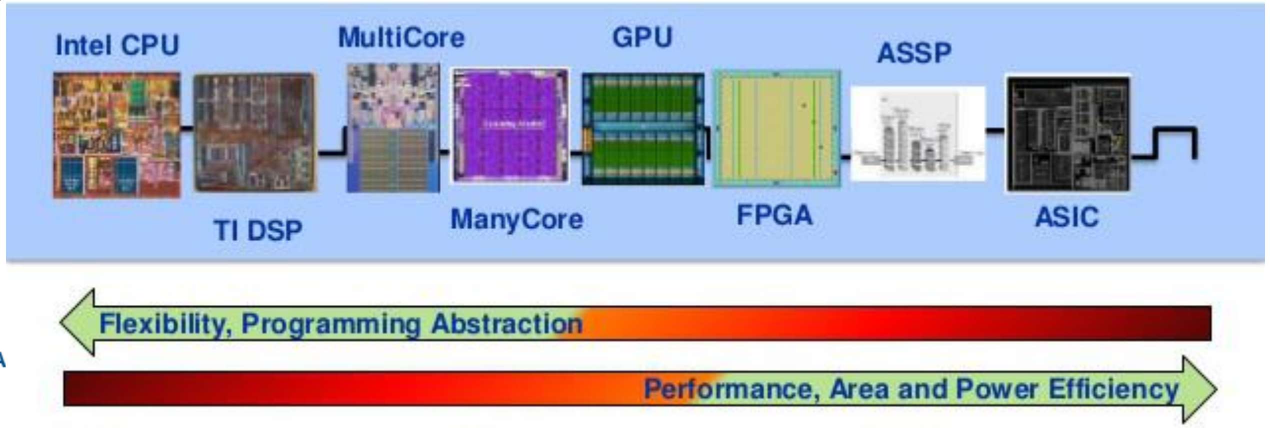
Atlas



GridTools



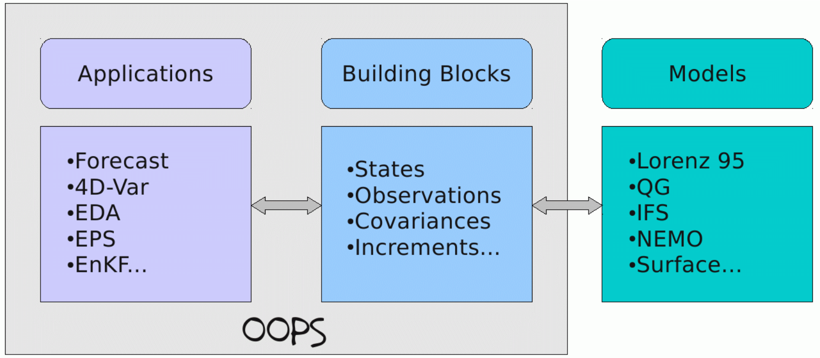
Processors

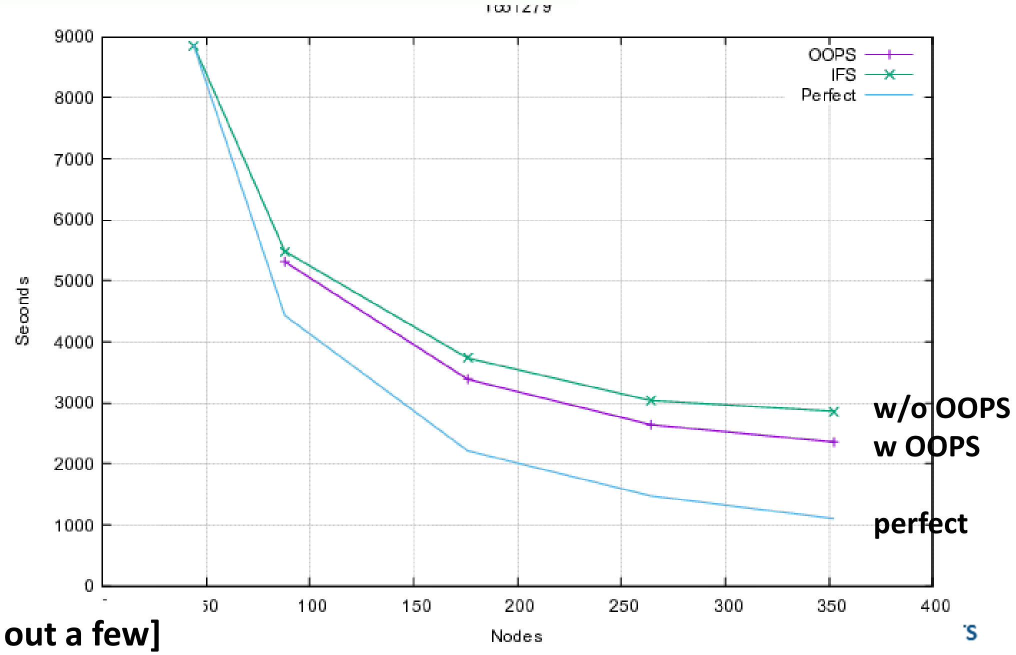


Far-hanging fruits: Algorithms – programming - hardware

Algorithmic flexibility equally applicable to:

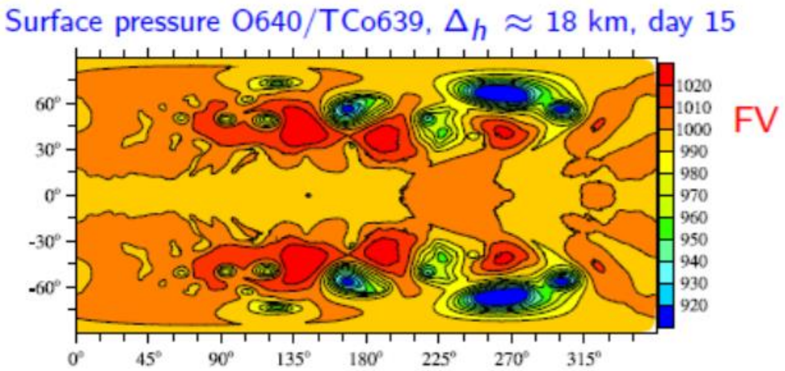
Data Assimilation

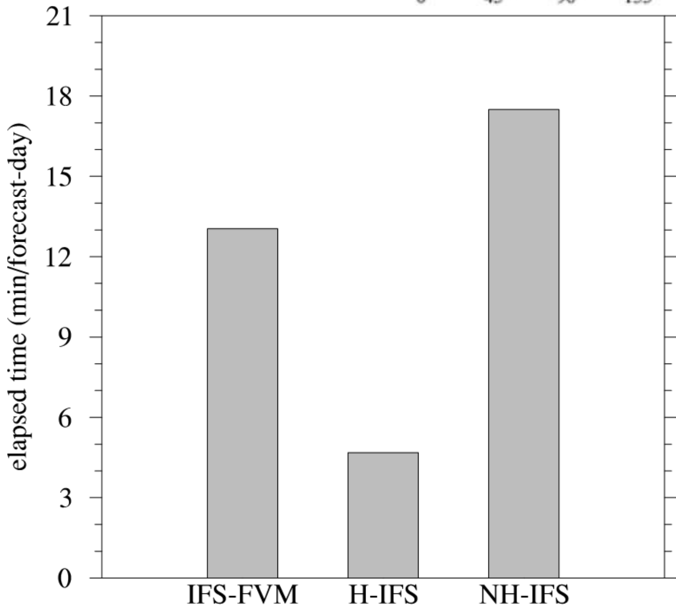


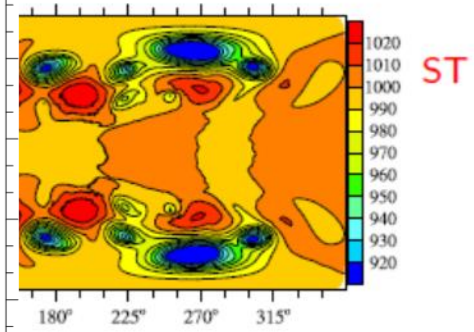


[Too many to single out a few]

Forecasting





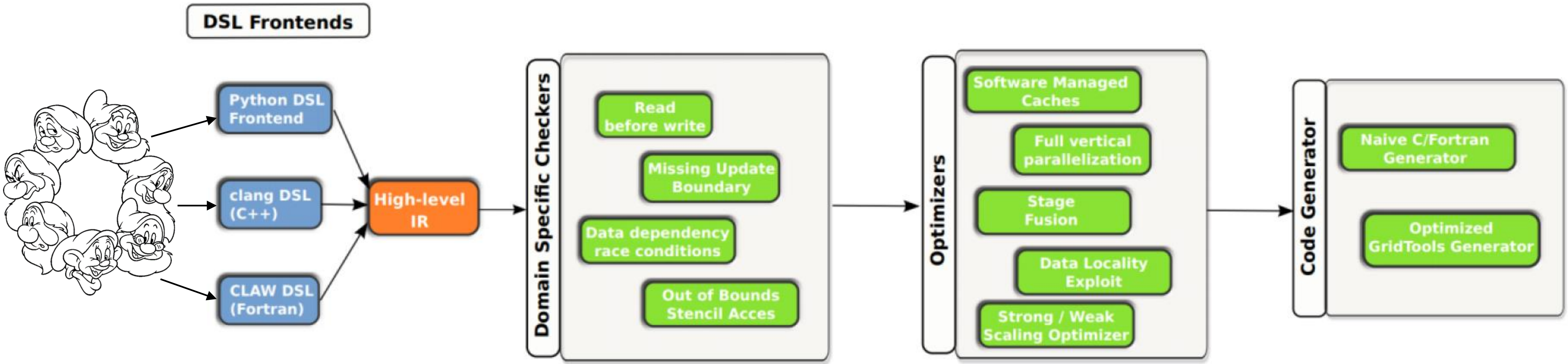


O1280/TCo1279 and L137 using dry dycore on 350 nodes of ECMWF's Cray XC40

[Christoph Kühnlein, Piotr Smolarkiewicz

Far-hanging fruits: Algorithms – programming - hardware

The ESCAPE 2 DSL tool-chain …



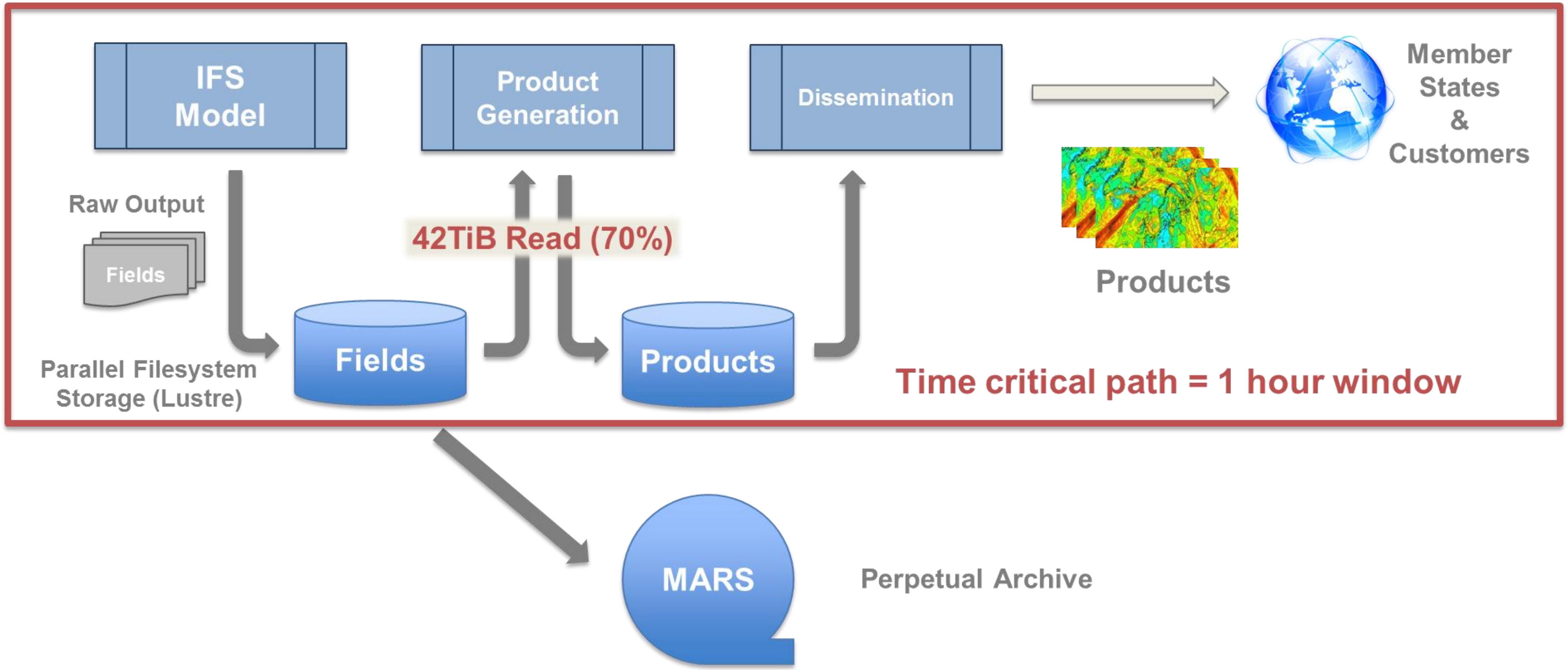
… supported by the EURO EXA toolchain:

1. Generate single-column abstraction code for physics dwarfs using Loki (=Python code transformation)
2. Generate GPU-code using CLAW
3. Generate prototype C-kernel for initial FPGA porting

[Michael Lange, Olivier Marsden]

Funded by the European Union

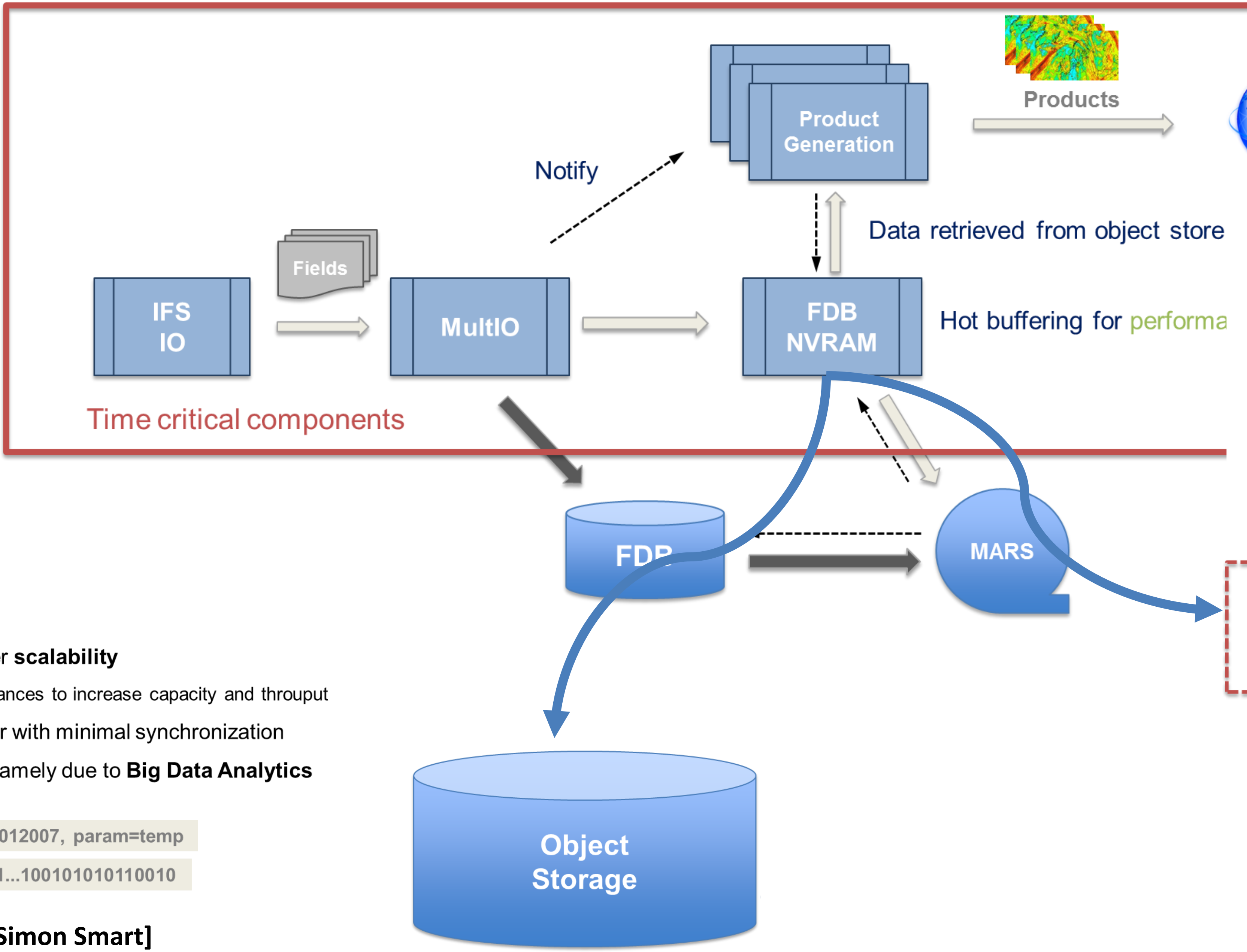
Examples of far-hanging fruit: Post-processing

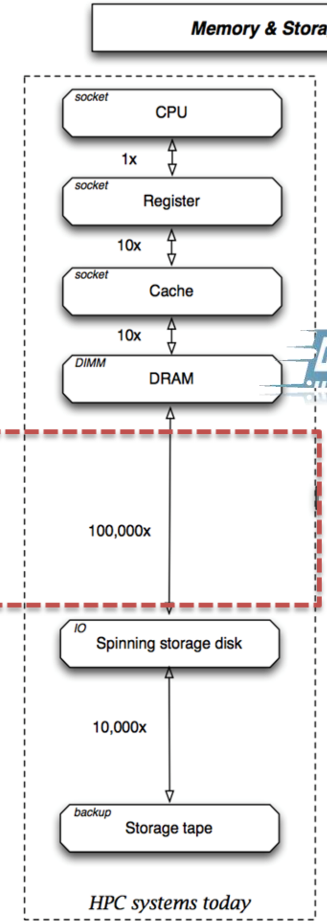


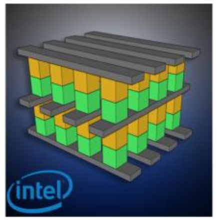
	Model	Model + I/O	Model + I/O + PGen
Nodes	2440	2776	2926
Run time [s]	5765	6749	7260
Relative	-	+ 17%	+ 26%

[Tiago Quintino, Simon Smart]

Examples of far-hanging fruit: Post-processing







- Key-Value stores offer **scalability**
  - Just add more instances to increase capacity and throuput
- **Transaction** behavior with minimal synchronization
- Growing popularity, namely due to **Big Data Analytics**

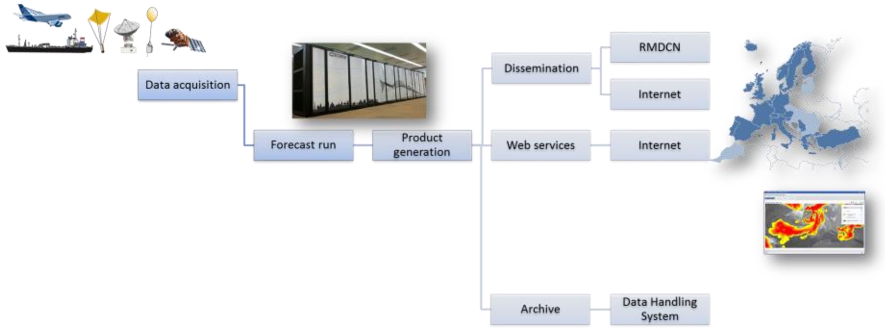
Key: date=12012007, param=temp

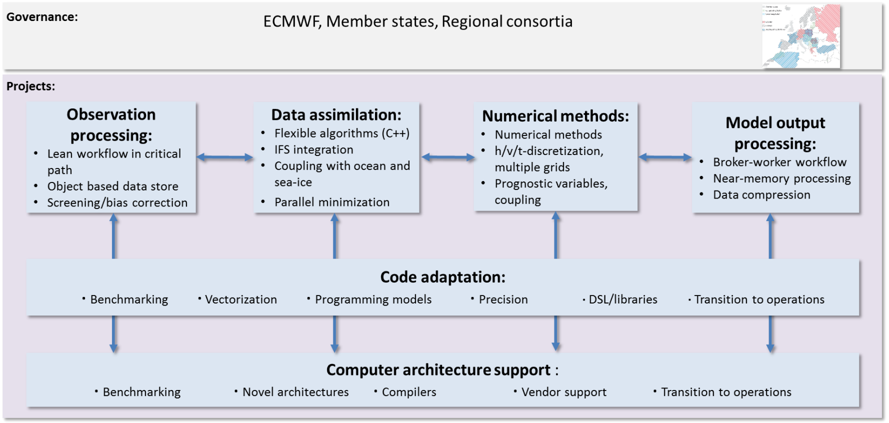
Value: 101001...100101010110010

[Tiago Quintino, Simon Smart]

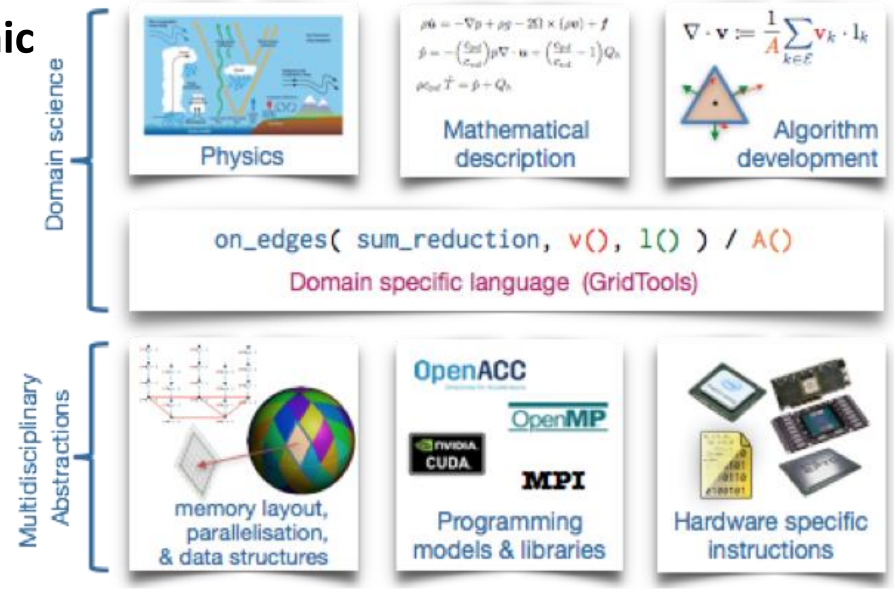
ECMWF Scalability Programme 2.0

Scalability Programme 1.0

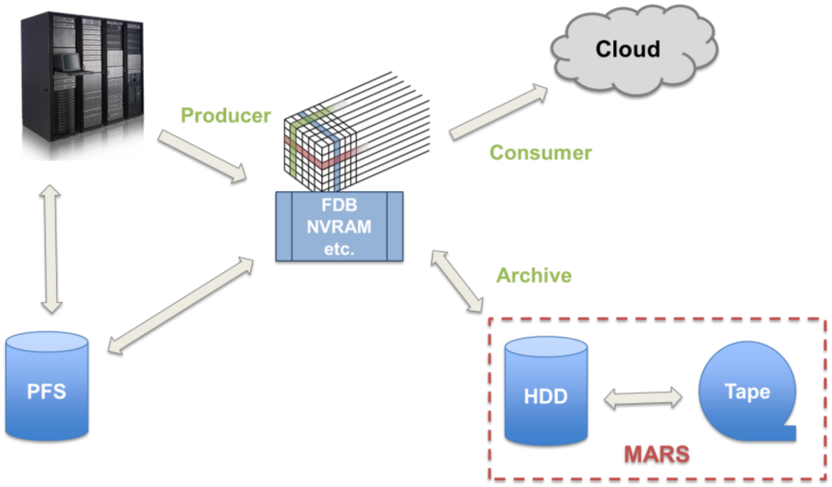




Co-development of algorithmic options, programming and hardware:



Co-development of flexible workflows, object data stores and hardware (cloud aware):



There is an opportunity to take this to the extreme!

FET FLAGSHIPS

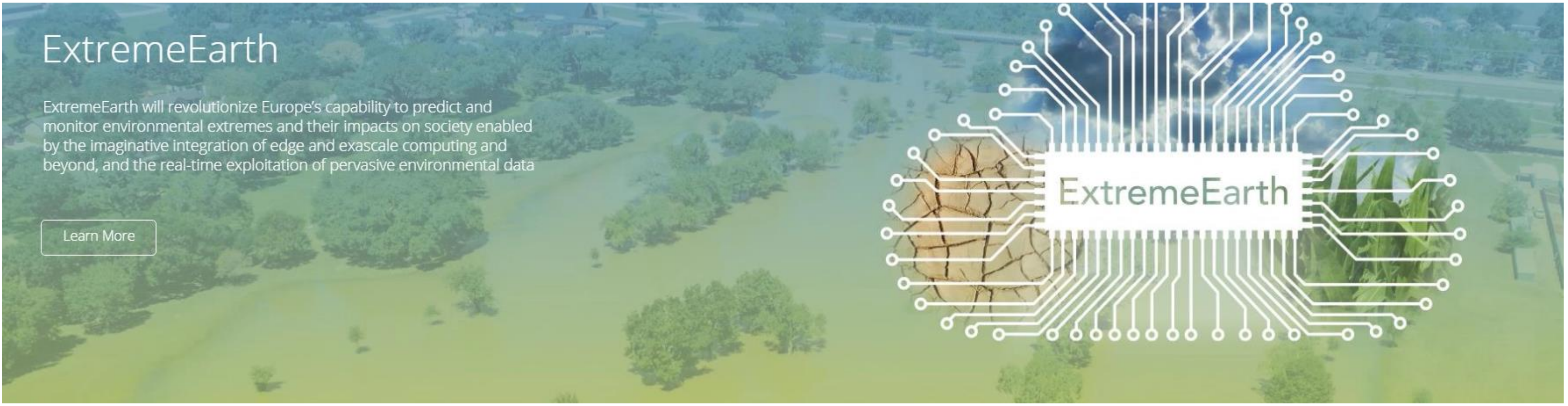
Human Brain Project

GRAPHENE FLAGSHIP

Future and Emerging Technology Flagships are:

"… science- and technology-driven, large-scale, multidisciplinary research initiatives built around a visionary unifying goal ... tackle grand science and technology challenges … strong and broad basis for future innovation and economic exploitation … novel benefits for society of a potential high impact … long-term and sustained effort."

The *ExtremeEarth* proposal: **www.extremeearth.eu**



ECMWF — European Centre for Medium-Range Weather Forecasts
European Commission — Joint Research Centre
Max-Planck-Institute for Biogeochemistry
University of Oxford
JÜLICH Forschungszentrum — Forschungszentrum Juelich GmbH
ETH zürich — Eidgenoessische Technische Hochschule Zuerich
cnrs — Centre National de la Recherche Scientifique
cmcc — Centro Euro-Mediterraneo per i Cambiamenti Climatici
netherlands eScience center — Netherlands eScience Center
Deltares — Enabling Delta Life — Deltares
DTU — Danish Technical University
BSC — Barcelona Supercomputing Center-Centro Nacional de Supercomputación
Climate Centre — Red Cross Red Crescent Climate Centre
UK Research and Innovation
METEO FRANCE — Meteo-France
Universiteit Utrecht — University Utrecht
Istituto Nazionale di Geofisica e Vulcanologia
UNIVERSITY OF HELSINKI — University Helsinki

Co-Design

Science:

- Weather and Climate extremes
- Volcanoes
- Earthquakes

Disasters & risks:

- Hydrology and water
- Energy
- Food and agriculture
- Health
- Geo-engineering

Advanced mathematics & algorithms

Multi-scale/multi-physics models

Portable and performant science code

Domain specific computing framework

End-to-end demonstrators

Technology:

- Numerical modelling
- Data assimilation and fusion
- Artificial intelligence methods
- Programming models
- Extreme-scale computing
- Extreme-scale data handling and storage
- Workflows and visualization

ExtremeEarth

Ultra high-resolution Earth-system physics-impact modelling capability

Extreme-scale Earth-system data management capability

Extreme-scale Earth-system computing capability

Interactive Earth-system information system capability

Future forecasting

Incremental:

```
do while (skill .ne. good_enough)
        model%resolution          = model%resolution          / model%dresolution
        model%complexity          = model%complexity          * model%dcomplexity
        ensemble%size             = ensemble%size             * ensemble%dsize
        downstream%application = downstream%application + 1

        call performance (model, ensemble, downstream, speed)
        call translate   (model, ensemble, downstream, speed, software, hardware)

        do while (speed .ne. fast_enough)
                call add_funding      (bucks, software, hardware)
                call add_optimization (software, hardware, speed)
                call add_processors   (software, hardware, speed)

                if (bucks .gt. budget) abort
        end do
        call science (model, ensemble, downstream, skill)
end do
```

Radical:

```
call extreme_earth
```