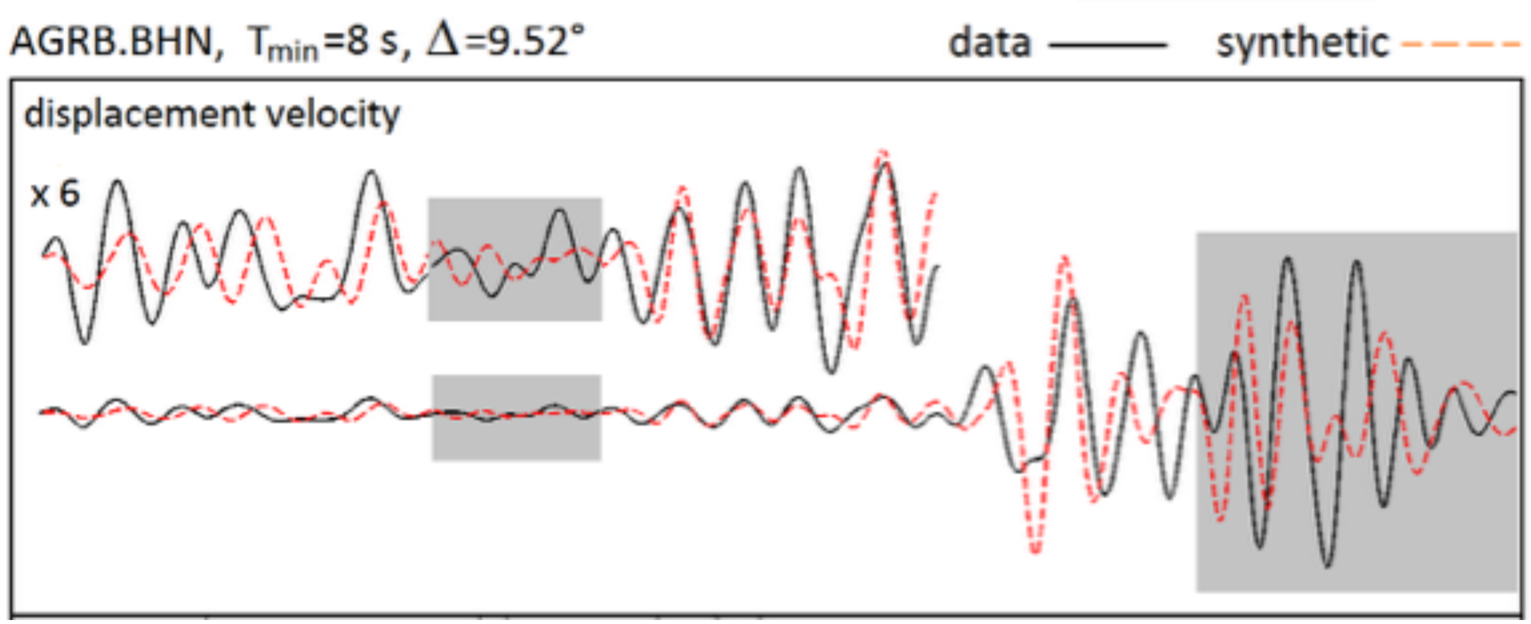
ETH zürich

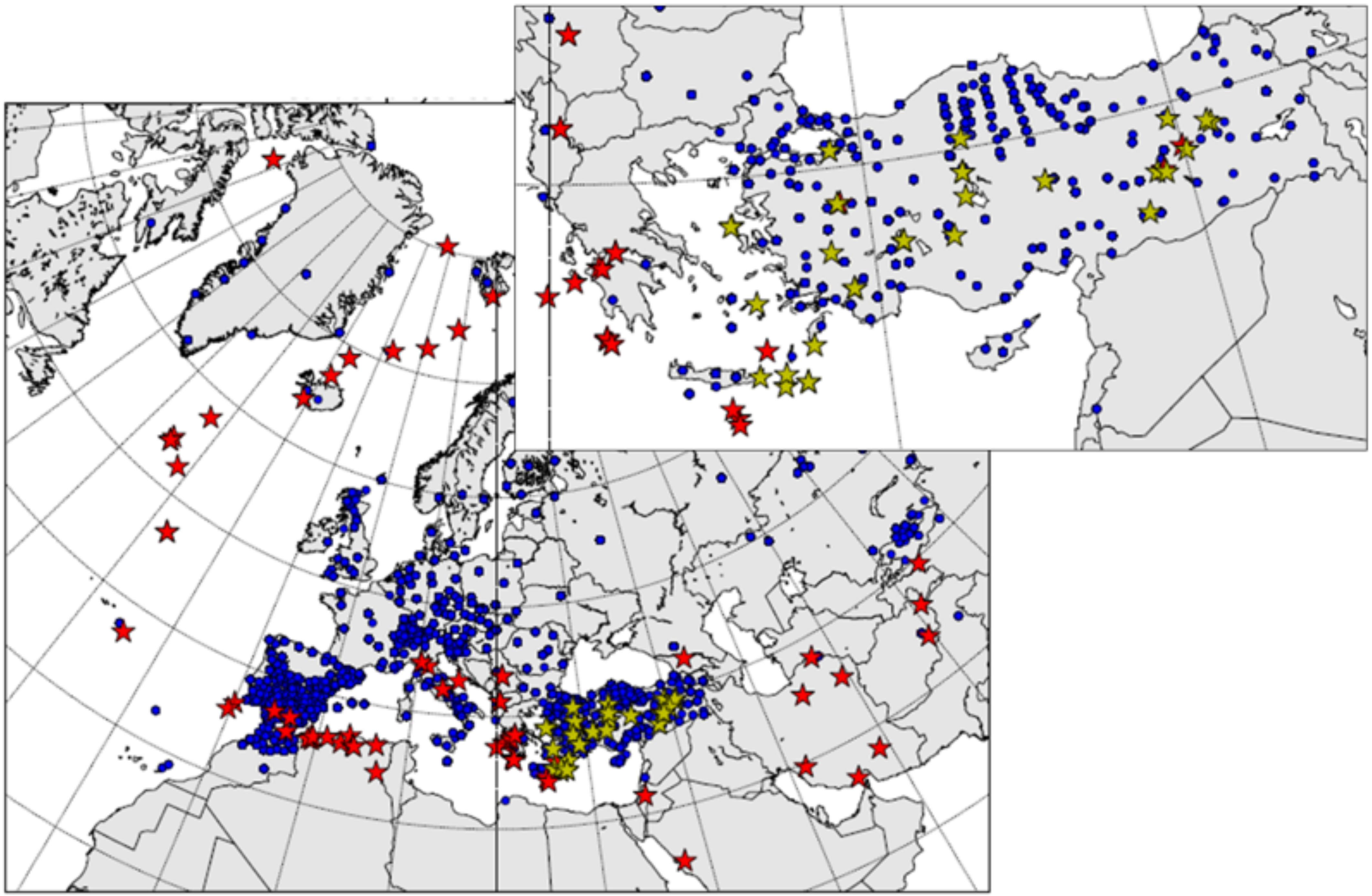
Piz Daint: a modern research infrastructure for computational science

Thomas C. Schulthess



source: A. Fichtner, ETH Zurich

Data from many stations and earthquakes



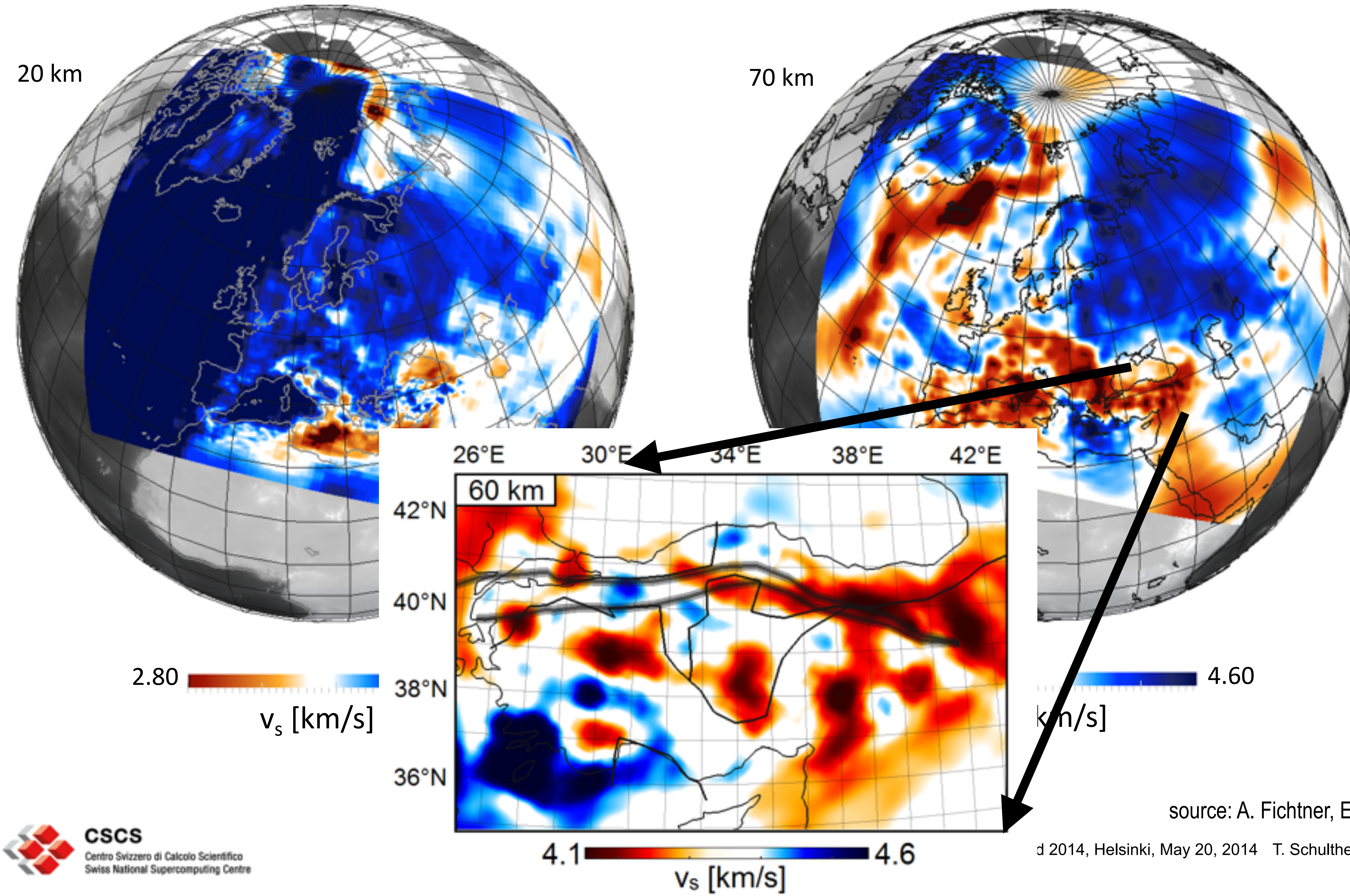
★ epicentres in the continent-wide data set (84)

● station

★ epicentres in the Anatolian data set (29)

source: A. Fichtner, ETH Zurich

Very large simulations allow inverting large data sets to generate high-resolution model or earth's mantle



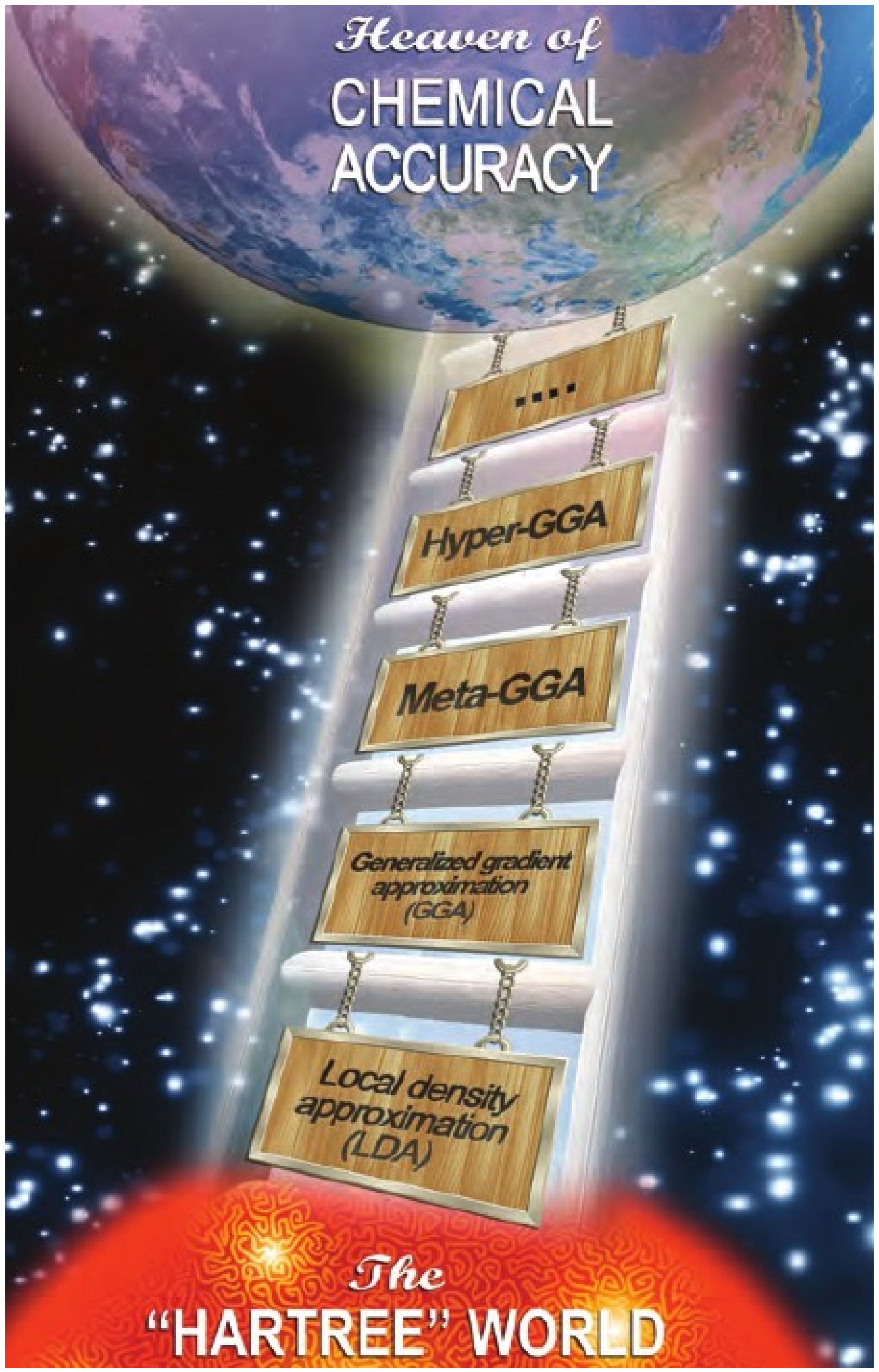
source: A. Fichtner, ETH Zurich

Ice floats!

first early science result on “Piz Daint” (Joost VandeVondele, Jan. 2014)

high-accuracy quantum simulation produce correct results for water

Jacobs ladder of chemical accuracy (J. Perdew)



Full many-body Schrödinger Equation:

$$w(\xi_i, \xi_j) = \frac{e^2}{|\vec{r}_i - \vec{r}_j|}$$

$$\xi = \{\vec{r}, \sigma\}$$

$$(\mathbb{H} - E)\Psi(\xi_1, ..., \xi_N) = 0$$

$$\mathbb{H} = \sum_{i=1}^{N} \left(-\frac{\hbar^2}{2m}\nabla_i^2 + v(\xi_i) \right) + \frac{1}{2} \sum_{i \neq j = 1}^{N} w(\xi_i, \xi_j)$$

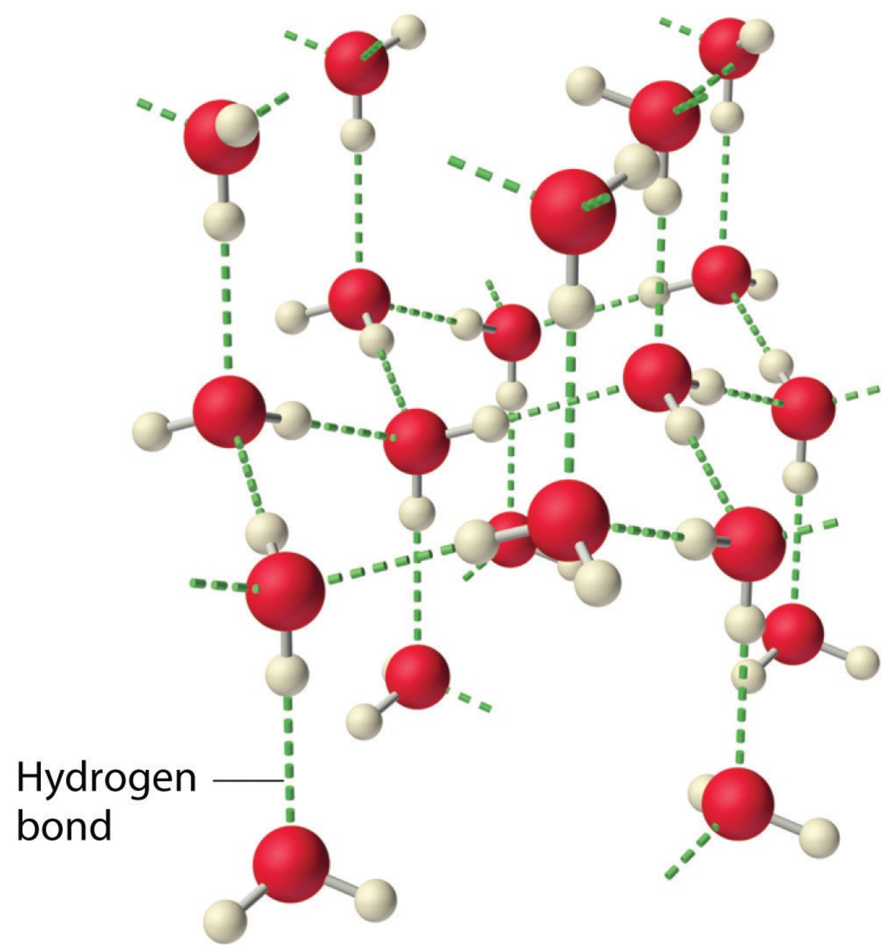
Ice floats with new "rung 5" simulations using MP2-based simulations with CP2K (VandeVondele & Hutter)

Ice didn't float with previous simulations using "rung 4" hybrid functionals

Kohn-Sham Equation with Local Density Approximation:

$$\left(-\frac{\hbar^2}{2m}\nabla^2 + v_{\mathrm{LDA}}(\vec{r}) \right) \psi_i(\vec{r}) = \epsilon_i \psi_i(\vec{r})$$

Modelling interactions between water requires quantum simulations with extreme accuracy



Energy scales

total energy: -76.438… a.u.
hydrogen bonds: ~0.0001 a.u.

required accuracy: 99.9999%

source: J. VandeVondele, ETH Zurich

Pillars of the scientific method

Mathematics / Simulation

(1) Synthesis of models and data: recognising characteristic features of complex systems with calculations of limited accuracy (e.g. inverse problems)

(2) Solving theoretical problems with high precision: complex structures emerge from simple rules (natural laws), more accurate predictions from “beautiful” theory (in the Penrose sense)

Theory (models)

Experiment (data)

Pillars of the scientific method

Mathematics / Simulation

(1) Synthesis of models and data: recognising characteristic features of complex systems with calculations of limited accuracy (e.g. inverse problems)

(2) Solving theoretical problems with high precision: complex structures emerge from simple rules (natural laws), more accurate predictions from "beautiful" theory

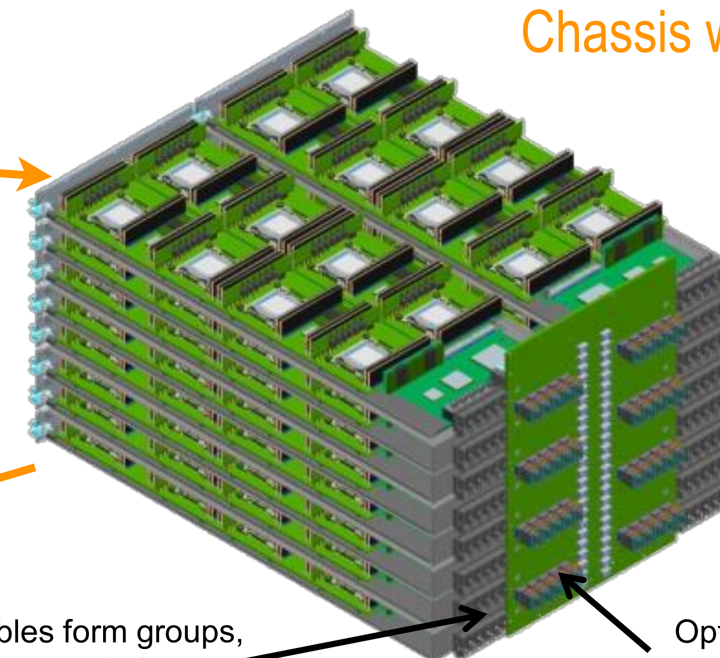
Note the changing role of high-performance computing: HPC is now an essential tool for science, used by all scientists (for better or worse), rather than being limited to the domain of applied mathematics and providing numerical solution to theoretical problems only few understand

CSCS' new flagship system "Piz Daint," and one of Europe's most powerful petascale supercomputers

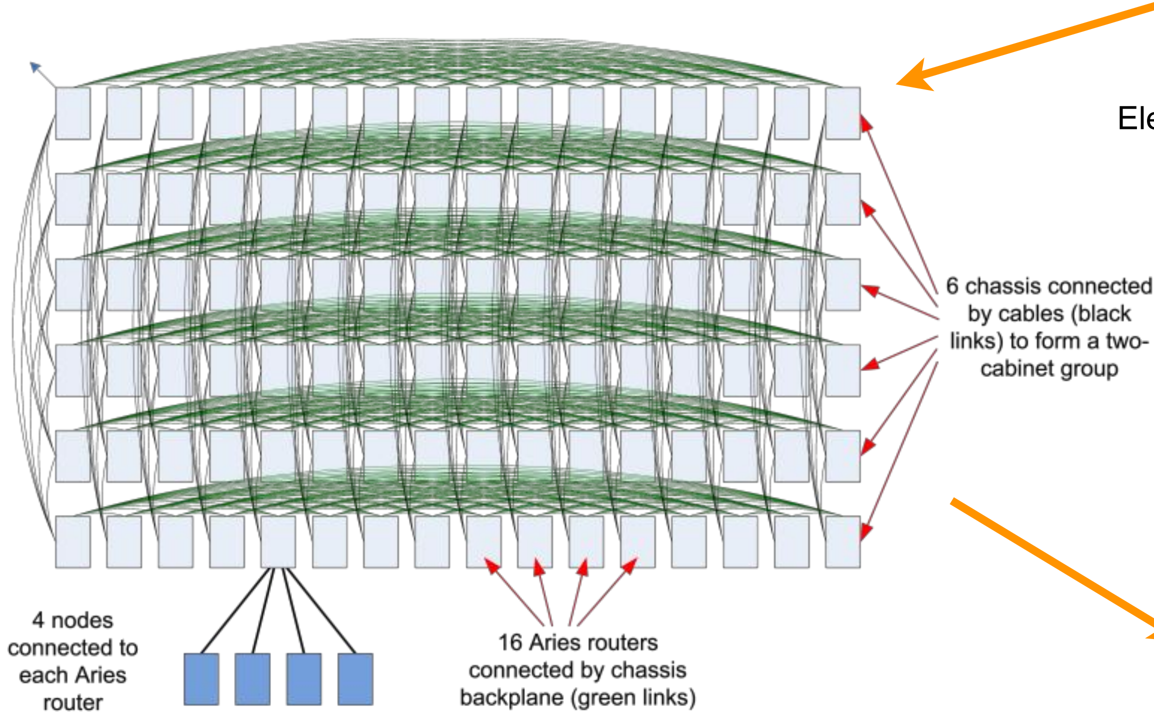
Presently the world's most energy efficient petascale supercomputer!

Hierarchical setup of Cray’s Cascade architecture

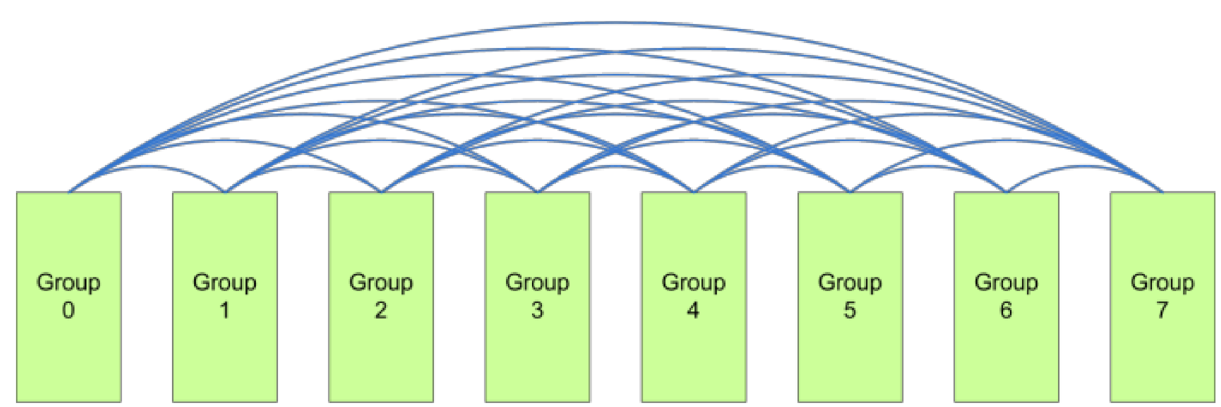
Blade with 4 dual socket nodes



Chassis with 16 blades



Electrical group (2 cabinets) with 6 chassis

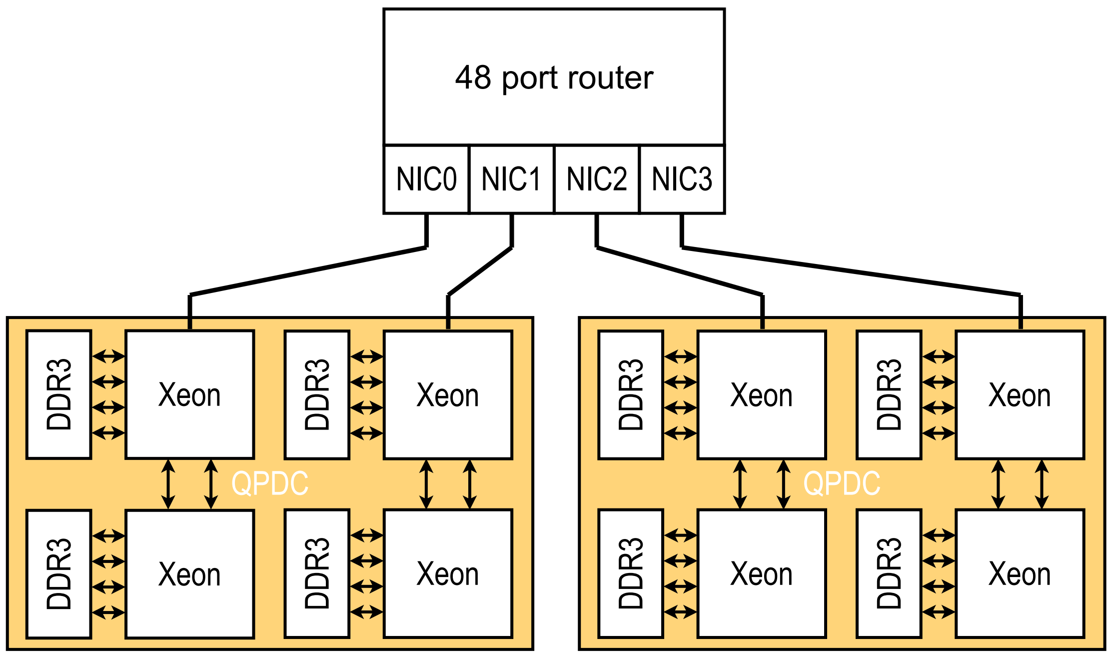


Cascade system with 8 electrical groups (16 cabinets)

Source: G. Fannes et al., SC’12 proceedings (2012)

Regular multi-core vs. hybrid-multi-core blades

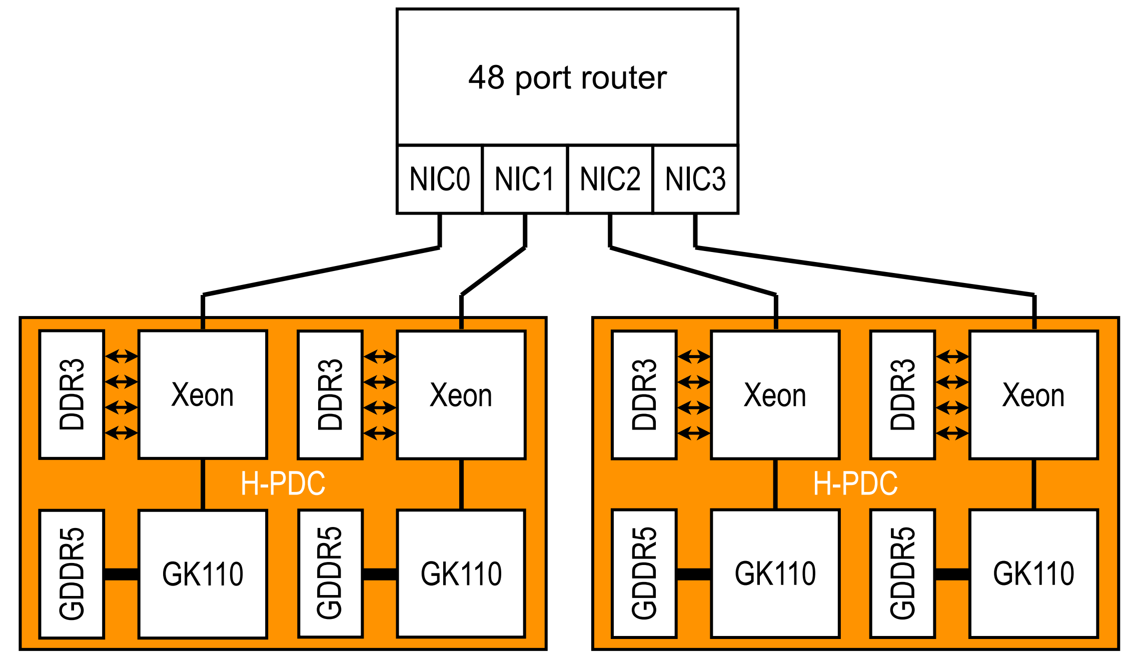
Initial Multi-core blade



4 nodes configured with:
> 2 Intel SandyBridge CPU
> 32 GB DDR3-1600 RAM

Peak performance of blade: 1.3 TFlops

Final hybrid CPU-GPU blade



4 nodes configured with:
> 1 Intel SandyBridge CPU
> 32 GB DDR3-1600 RAM
> 1 NVIDIA K20X GPU
> 6GB GDDR5 memory

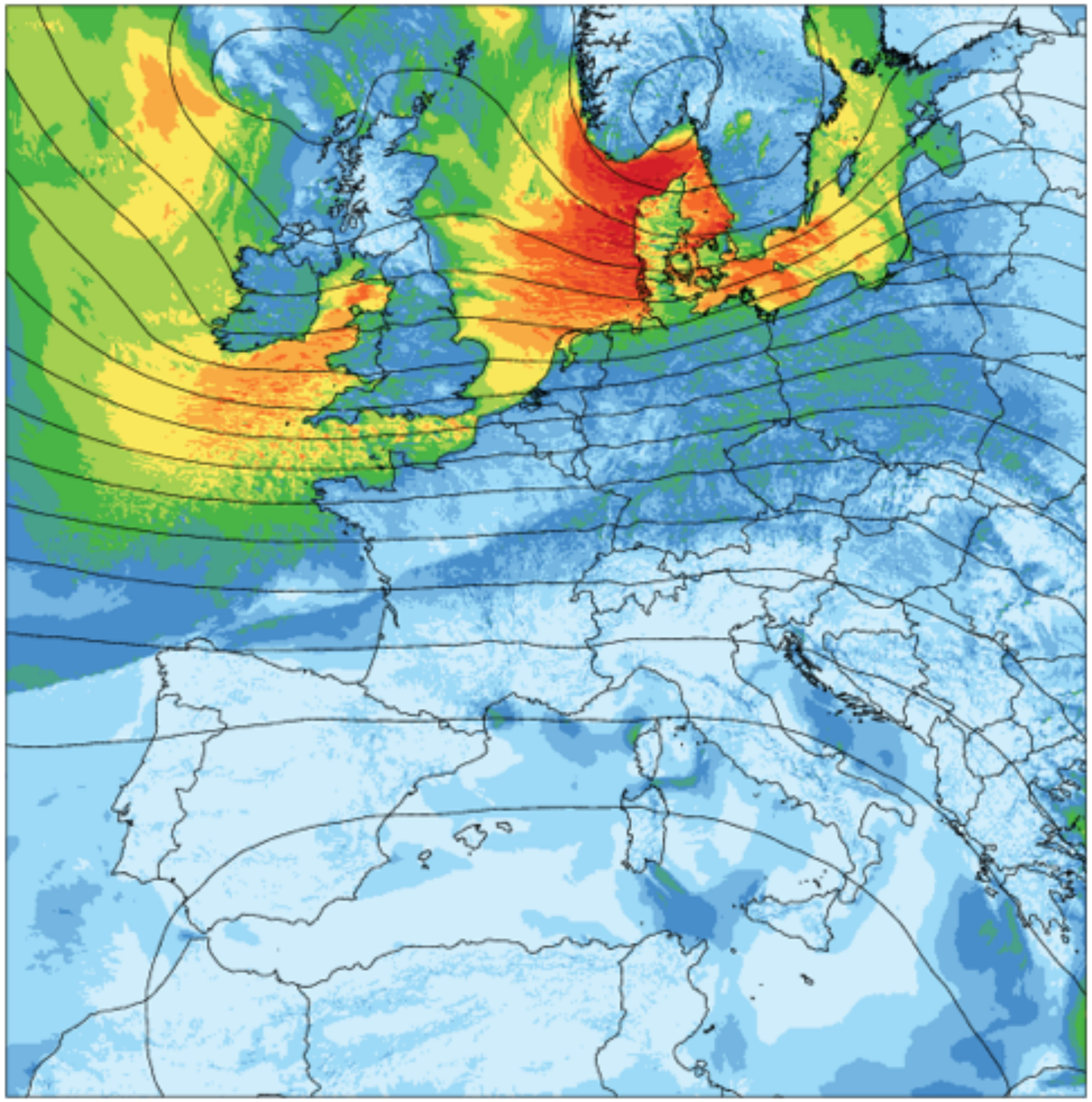
Peak performance of blade: 5.9 TFlops

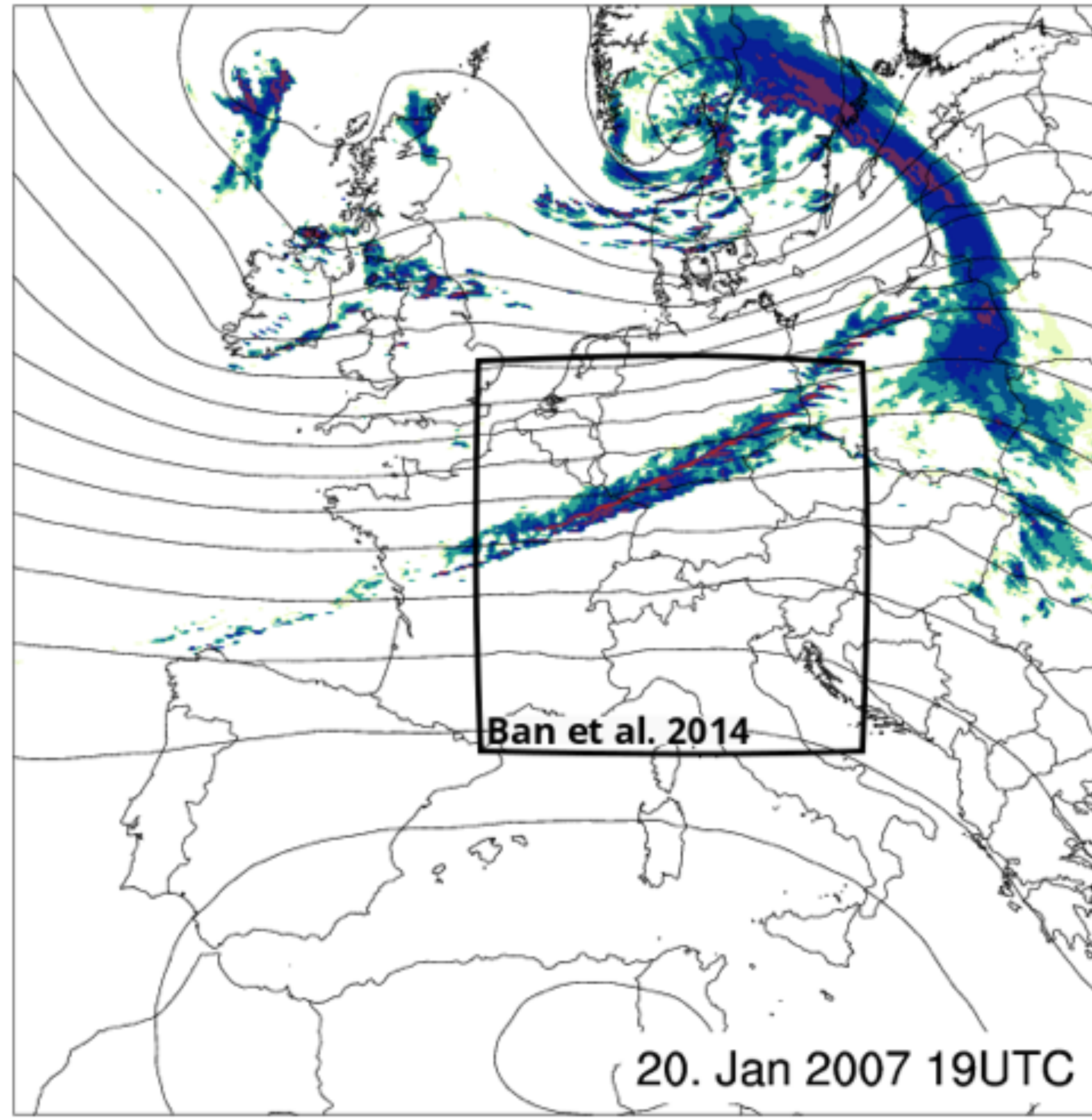
A brief history of "Piz Daint"

- Installed 12 cabinets with dual-SandyBridge nodes in Oct./Nov. 2012
  - ~50% of final system size to test network at scale (lessons learned from Gemini & XE6)
- Rigorous evaluation of three node architectures based on 9 applications
  - dual-Xeon vs. Xeon/Xeon-Phi vs. Xeon/Kepler
  - joint study with Cray from Dec. 2011 through Nov. 2012
- Five applications were used for system design
  - CP2K, COSMO, SPECFEM-3D, GROMACS, Quantum ESPRESSO
- CP2K & COSMO-OPCODE were co-developed with the system
  - Moving performance goals: hybrid Cray XC30 has to beat the regular XC30 by 1.5x
- Upgrade to 28 cabinets with hybrid CPU-GPU nodes in Oct./Nov. 2013
- Accepted in Dec. 2013
- Early Science with fully performance hybrid nodes: January through March 2014

Cloud-Resolving Simulation of Winter Storm Kyrill

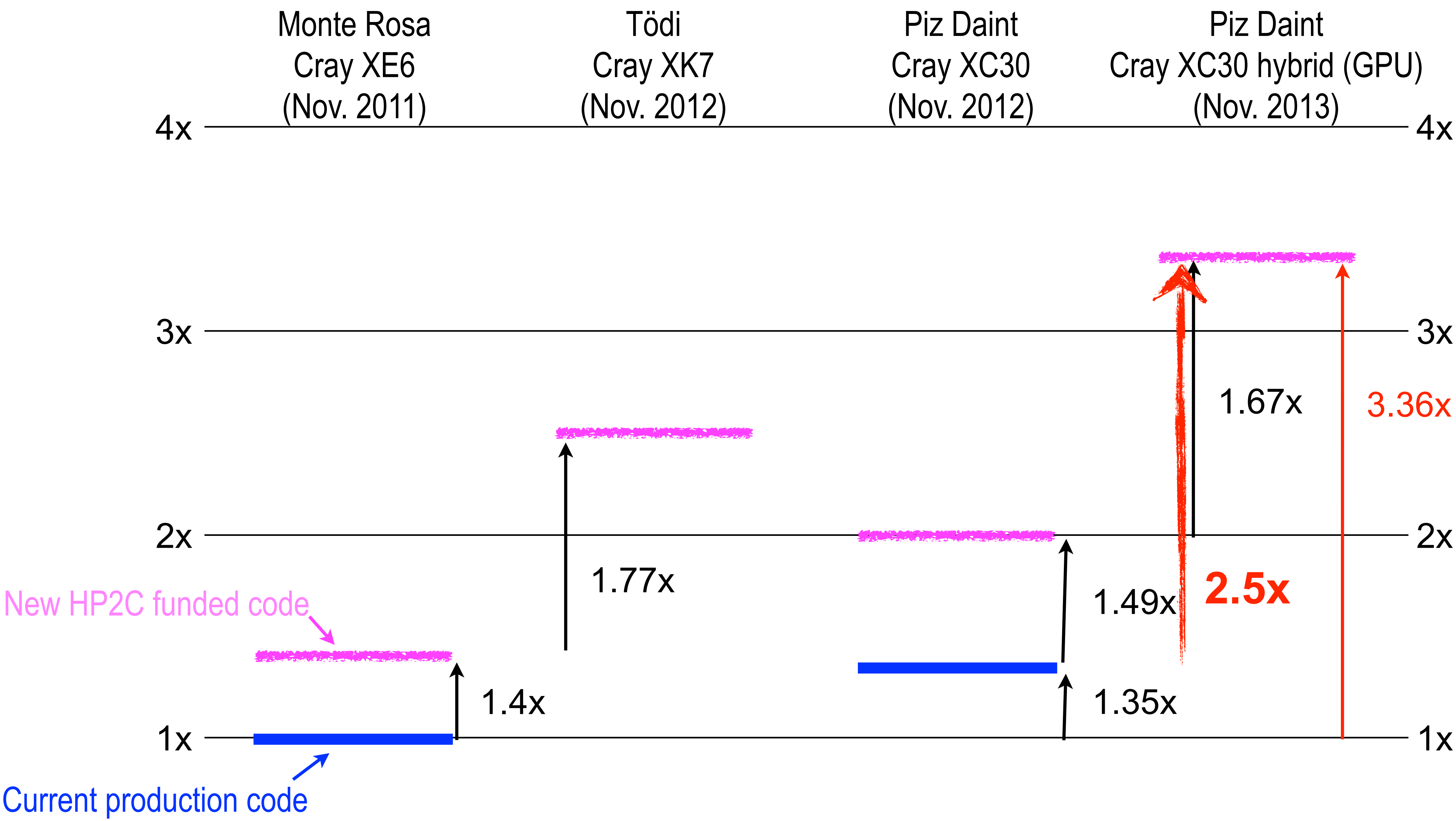
David Leutwyler, Oliver Fuhrer, Christoph Schär, Andrea Arteaga, Isabelle Bey, Mauro Bianco, Ben Cumming, Tobias Gysi, Xavier Lapillonne, Daniel Lüthi, Carlos Osuna, Anne Roches, Thomas Schulthess



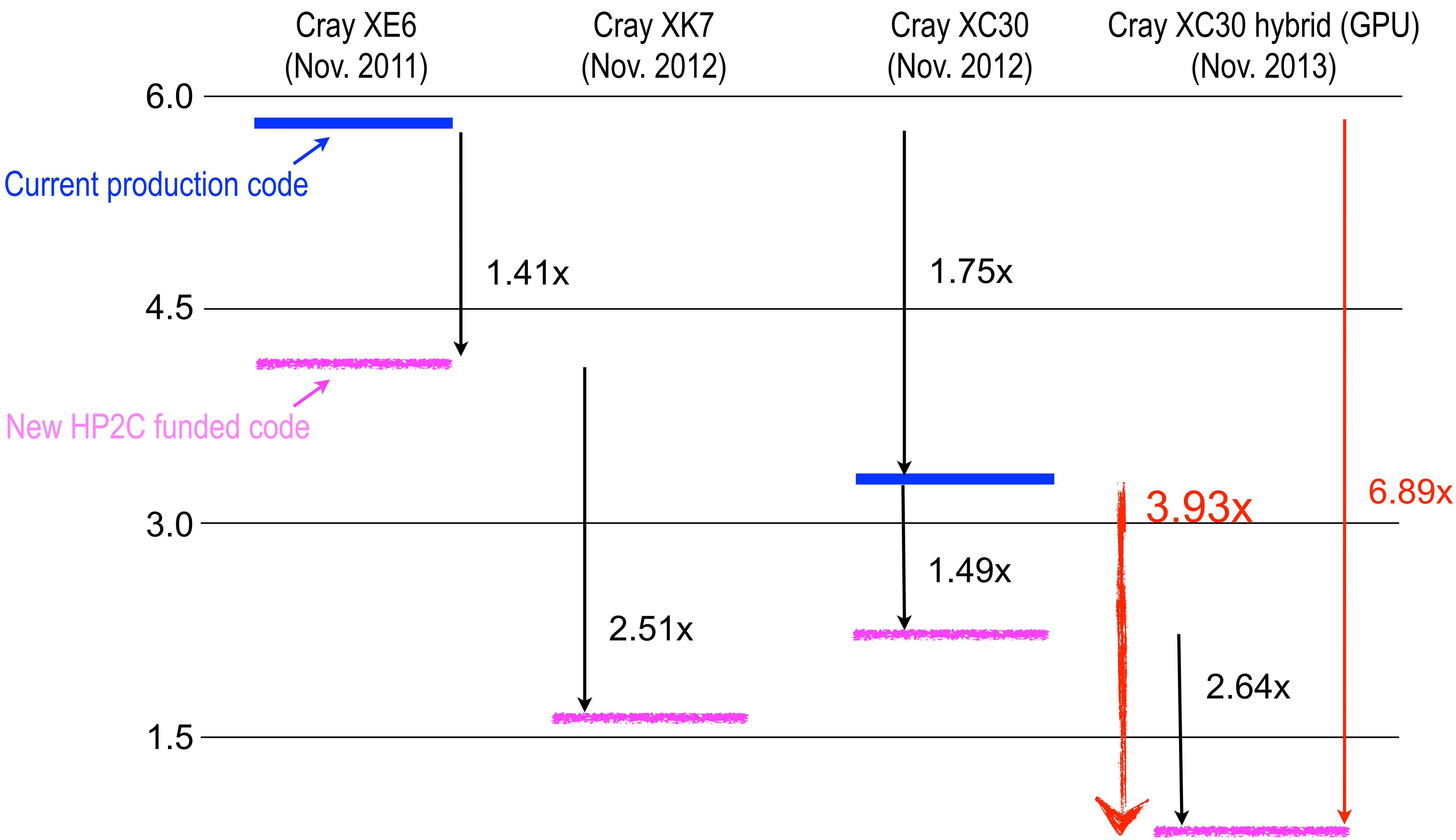


source: David Leutwyler

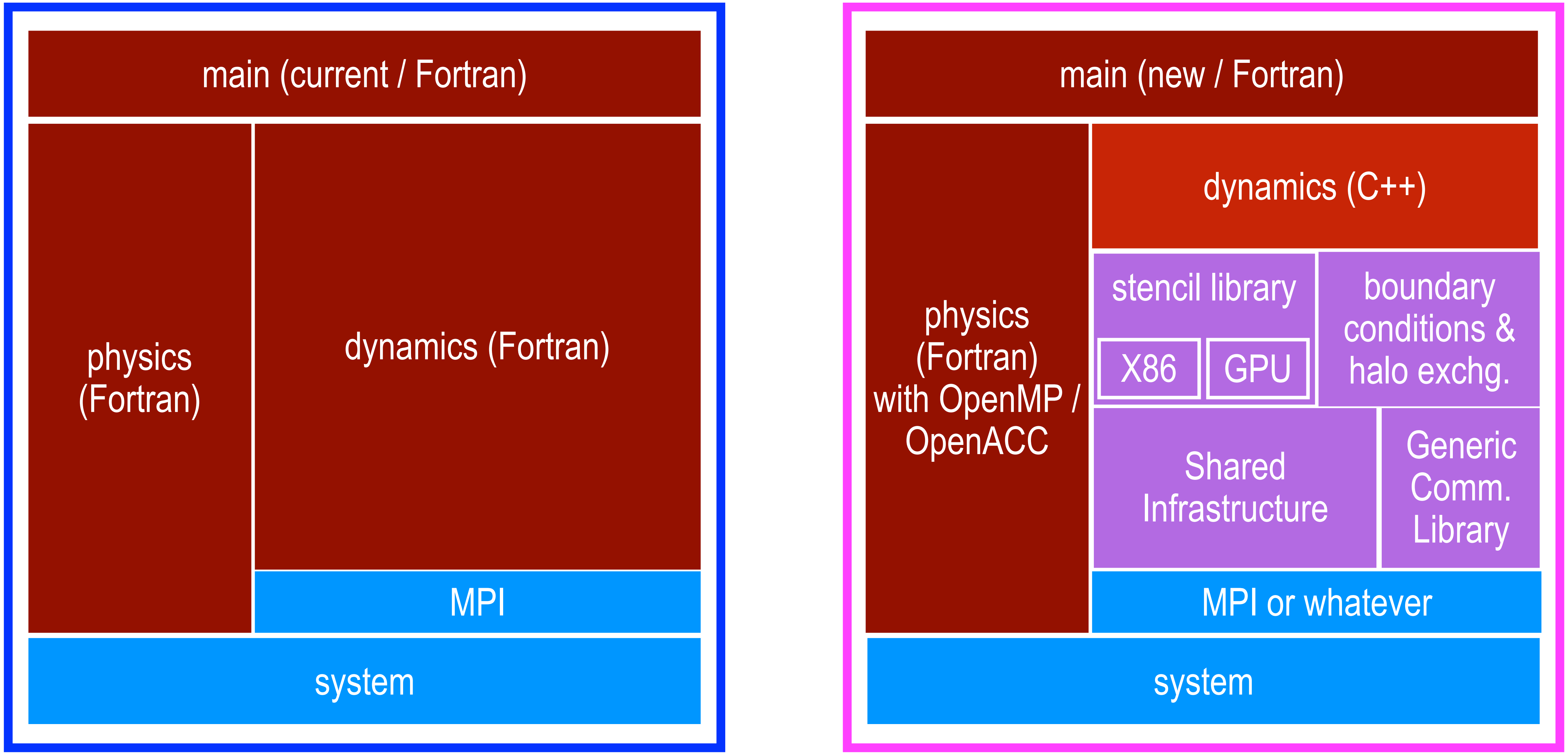
Speedup of the full COSMO-2 production problem (apples to apples with 33h forecast of Meteo Swiss)



Energy to solution (kWh / ensemble member)



COSMO: current and new (HP2C developed) code



ETH zürich

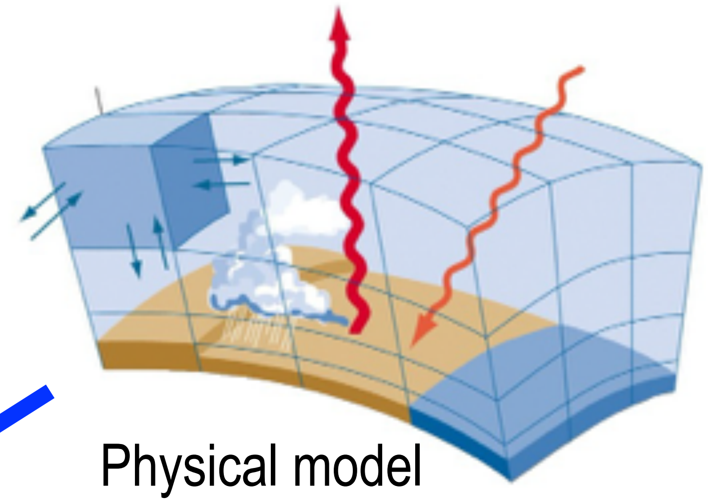
$$
\text{velocities}\begin{cases}
\dfrac{\partial u}{\partial t} = -\left\{\dfrac{1}{a\cos\varphi}\dfrac{\partial E_h}{\partial\lambda} - vV_a\right\} - \dot{\zeta}\dfrac{\partial u}{\partial\zeta} - \dfrac{1}{\rho a\cos\varphi}\left(\dfrac{\partial p'}{\partial\lambda} - \dfrac{1}{\sqrt{\gamma}}\dfrac{\partial p_0}{\partial\lambda}\dfrac{\partial p'}{\partial\zeta}\right) + M_u \\[2ex]
\dfrac{\partial v}{\partial t} = -\left\{\dfrac{1}{a}\dfrac{\partial E_h}{\partial\varphi} + uV_a\right\} - \dot{\zeta}\dfrac{\partial v}{\partial\zeta} - \dfrac{1}{\rho a}\left(\dfrac{\partial p'}{\partial\varphi} - \dfrac{1}{\sqrt{\gamma}}\dfrac{\partial p_0}{\partial\varphi}\dfrac{\partial p'}{\partial\zeta}\right) + M_v \\[2ex]
\dfrac{\partial w}{\partial t} = -\left\{\dfrac{1}{a\cos\varphi}\left(u\dfrac{\partial w}{\partial\lambda} + v\cos\varphi\dfrac{\partial w}{\partial\varphi}\right)\right\} - \dot{\zeta}\dfrac{\partial w}{\partial\zeta} + \dfrac{g}{\sqrt{\gamma}}\dfrac{\rho_0}{\rho}\dfrac{\partial p'}{\partial\zeta} + M_w + g\dfrac{\rho_0}{\rho}\left\{\dfrac{(T-T_0)}{T} - \dfrac{T_0 p'}{T p_0} + \left(\dfrac{R_v}{R_d} - 1\right)q^v - q^l - q^f\right\}
\end{cases}
$$

$$
\text{pressure} \quad \frac{\partial p'}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial p'}{\partial\lambda} + v\cos\varphi\frac{\partial p'}{\partial\varphi}\right)\right\} - \dot{\zeta}\frac{\partial p'}{\partial\zeta} + g\rho_0 w - \frac{c_{pd}}{c_{vd}}pD
$$

$$
\text{temperature} \quad \frac{\partial T}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial T}{\partial\lambda} + v\cos\varphi\frac{\partial T}{\partial\varphi}\right)\right\} - \dot{\zeta}\frac{\partial T}{\partial\zeta} - \frac{1}{\rho c_{vd}}pD + Q_T
$$

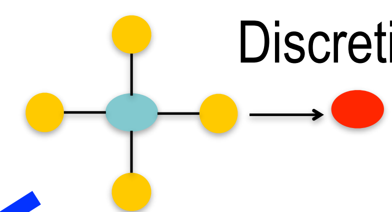
$$
\text{water}\begin{cases}
\dfrac{\partial q^v}{\partial t} = -\left\{\dfrac{1}{a\cos\varphi}\left(u\dfrac{\partial q^v}{\partial\lambda} + v\cos\varphi\dfrac{\partial q^v}{\partial\varphi}\right)\right\} - \dot{\zeta}\dfrac{\partial q^v}{\partial\zeta} - (S^l + S^f) + M_{q^v} \\[2ex]
\dfrac{\partial q^{l,f}}{\partial t} = -\left\{\dfrac{1}{a\cos\varphi}\left(u\dfrac{\partial q^{l,f}}{\partial\lambda} + v\cos\varphi\dfrac{\partial q^{l,f}}{\partial\varphi}\right)\right\} - \dot{\zeta}\dfrac{\partial q^{l,f}}{\partial\zeta} - \dfrac{g}{\sqrt{\gamma}}\dfrac{\rho_0}{\rho}\dfrac{\partial P_{l,f}}{\partial\zeta} + S^{l,f} + M_{q^{l,f}}
\end{cases}
$$

$$
\text{turbulence} \quad \frac{\partial e_t}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial e_t}{\partial\lambda} + v\cos\varphi\frac{\partial e_t}{\partial\varphi}\right)\right\} - \dot{\zeta}\frac{\partial e_t}{\partial\zeta} + K_m^v\frac{g\rho_0}{\sqrt{\gamma}}\left\{\left(\frac{\partial u}{\partial\zeta}\right)^2 + \left(\frac{\partial v}{\partial\zeta}\right)^2\right\} + \frac{g}{\rho\theta_v}F^{\theta_v} - \frac{\sqrt{2}e_t^{3/2}}{\alpha_M l} + M_{e_t}
$$



Physical model

Mathematical description



Discretization / algorithm

Domain science (incl. applied mathematics)

```
lap(i,j,k) = -4.0 * data(i,j,k) +
    data(i+1,j,k) + data(i-1,j,k) +
    data(i,j+1,k) + data(i,j-1,k);
```

Code / implementation

Code compilation

"Port" serial code to supercomputers
- > vectorize
- > parallelize
- > petascaling
- > exascaling
- > ...

Computer engineering (& computer science)

A given supercomputer

CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zürich

velocities

$$\frac{\partial u}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\frac{\partial E_h}{\partial\lambda} - vV_a\right\} - \dot{\zeta}\frac{\partial u}{\partial\zeta} - \frac{1}{\rho a\cos\varphi}\left(\frac{\partial p'}{\partial\lambda} - \frac{1}{\sqrt{\gamma}}\frac{\partial p_0}{\partial\lambda}\frac{\partial p'}{\partial\zeta}\right) + M_u$$

$$\frac{\partial v}{\partial t} = -\left\{\frac{1}{a}\frac{\partial E_h}{\partial\varphi} + uV_a\right\} - \dot{\zeta}\frac{\partial v}{\partial\zeta} - \frac{1}{\rho a}\left(\frac{\partial p'}{\partial\varphi} - \frac{1}{\sqrt{\gamma}}\frac{\partial p_0}{\partial\varphi}\frac{\partial p'}{\partial\zeta}\right) + M_v$$

$$\frac{\partial w}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial w}{\partial\lambda} + v\cos\varphi\frac{\partial w}{\partial\varphi}\right)\right\} - \dot{\zeta}\frac{\partial w}{\partial\zeta} + \frac{g}{\sqrt{\gamma}}\frac{\rho_0}{\rho}\frac{\partial p'}{\partial\zeta} + M_w + g\frac{\rho_0}{\rho}\left\{\frac{(T-T_0)}{T} - \frac{T_0 p'}{T p_0} + \left(\frac{R_v}{R_d} - 1\right)q^v - q^l - q^f\right\}$$

pressure

$$\frac{\partial p'}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial p'}{\partial\lambda} + v\cos\varphi\frac{\partial p'}{\partial\varphi}\right)\right\} - \dot{\zeta}\frac{\partial p'}{\partial\zeta} + g\rho_0 w - \frac{c_{pd}}{c_{vd}}pD$$

temperature

$$\frac{\partial T}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial T}{\partial\lambda} + v\cos\varphi\frac{\partial T}{\partial\varphi}\right)\right\} - \dot{\zeta}\frac{\partial T}{\partial\zeta} - \frac{1}{\rho c_{vd}}pD + Q_T$$

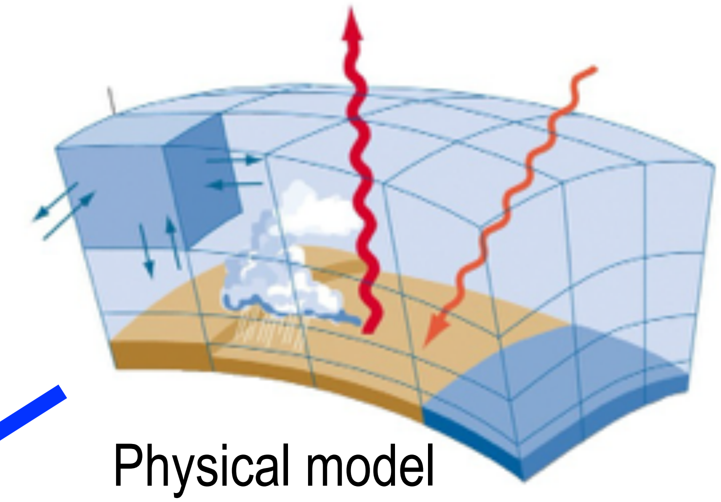
water

$$\frac{\partial q^v}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial q^v}{\partial\lambda} + v\cos\varphi\frac{\partial q^v}{\partial\varphi}\right)\right\} - \dot{\zeta}\frac{\partial q^v}{\partial\zeta} - (S^l + S^f) + M_{q^v}$$

$$\frac{\partial q^{l,f}}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial q^{l,f}}{\partial\lambda} + v\cos\varphi\frac{\partial q^{l,f}}{\partial\varphi}\right)\right\} - \dot{\zeta}\frac{\partial q^{l,f}}{\partial\zeta} - \frac{g}{\sqrt{\gamma}}\frac{\rho_0}{\rho}\frac{\partial P_{l,f}}{\partial\zeta} + S^{l,f} + M_{q^{l,f}}$$

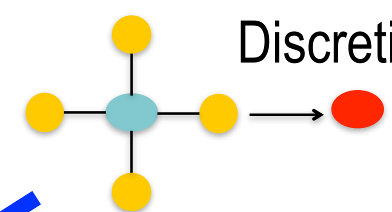
turbulence

$$\frac{\partial e_t}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial e_t}{\partial\lambda} + v\cos\varphi\frac{\partial e_t}{\partial\varphi}\right)\right\} - \dot{\zeta}\frac{\partial e_t}{\partial\zeta} + K_m^v\frac{g\rho_0}{\sqrt{\gamma}}\left\{\left(\frac{\partial u}{\partial\zeta}\right)^2 + \left(\frac{\partial v}{\partial\zeta}\right)^2\right\} + \frac{g}{\rho\theta_v}F^{\theta_v} - \frac{\sqrt{2}e_t^{3/2}}{\alpha_M l} + M_{e_t}$$



Physical model

Mathematical description



Discretization / algorithm

Domain science (incl. applied mathematics)

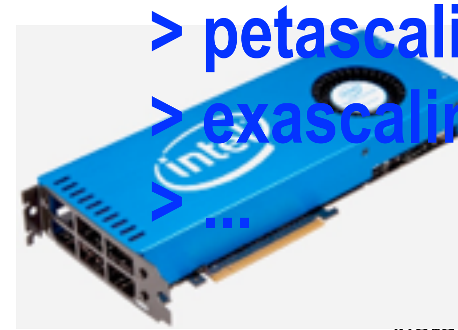
```
lap(i,j,k) = -4.0 * data(i,j,k) +
    data(i+1,j,k) + data(i-1,j,k) +
    data(i,j+1,k) + data(i,j-1,k);
```

Code / implementation

Code compilation

Architectural options / design

A given supercomputer



"Port" serial code to supercomputers

- > vectorize
- > parallelize
- > petascaling
- > exascaling
- > ...

Computer engineering (& computer science)

ETH zürich

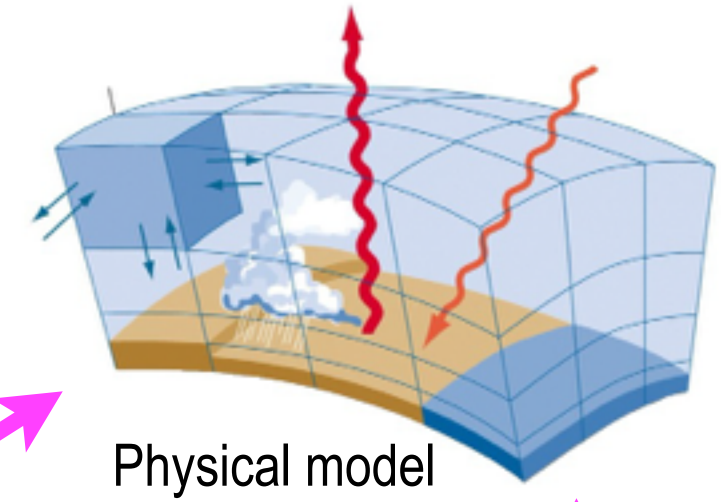
velocities
$$
\begin{cases}
\dfrac{\partial u}{\partial t} = -\left\{\dfrac{1}{a\cos\varphi}\dfrac{\partial E_h}{\partial\lambda} - vV_a\right\} - \dot\zeta\dfrac{\partial u}{\partial\zeta} - \dfrac{1}{\rho a\cos\varphi}\left(\dfrac{\partial p'}{\partial\lambda} - \dfrac{1}{\sqrt{\gamma}}\dfrac{\partial p_0}{\partial\lambda}\dfrac{\partial p'}{\partial\zeta}\right) + M_u \\
\dfrac{\partial v}{\partial t} = -\left\{\dfrac{1}{a}\dfrac{\partial E_h}{\partial\varphi} + uV_a\right\} - \dot\zeta\dfrac{\partial v}{\partial\zeta} - \dfrac{1}{\rho a}\left(\dfrac{\partial p'}{\partial\varphi} - \dfrac{1}{\sqrt{\gamma}}\dfrac{\partial p_0}{\partial\varphi}\dfrac{\partial p'}{\partial\zeta}\right) + M_v \\
\dfrac{\partial w}{\partial t} = -\left\{\dfrac{1}{a\cos\varphi}\left(u\dfrac{\partial w}{\partial\lambda} + v\cos\varphi\dfrac{\partial w}{\partial\varphi}\right)\right\} - \dot\zeta\dfrac{\partial w}{\partial\zeta} + \dfrac{g}{\sqrt{\gamma}}\dfrac{\rho_0}{\rho}\dfrac{\partial p'}{\partial\zeta} + M_w + g\dfrac{\rho_0}{\rho}\left\{\dfrac{(T-T_0)}{T} - \dfrac{T_0 p'}{T p_0} + \left(\dfrac{R_v}{R_d} - 1\right)q^v - q^l - q^f\right\}
\end{cases}
$$

pressure
$$
\dfrac{\partial p'}{\partial t} = -\left\{\dfrac{1}{a\cos\varphi}\left(u\dfrac{\partial p'}{\partial\lambda} + v\cos\varphi\dfrac{\partial p'}{\partial\varphi}\right)\right\} - \dot\zeta\dfrac{\partial p'}{\partial\zeta} + g\rho_0 w - \dfrac{c_{pd}}{c_{vd}}pD
$$

temperature
$$
\dfrac{\partial T}{\partial t} = -\left\{\dfrac{1}{a\cos\varphi}\left(u\dfrac{\partial T}{\partial\lambda} + v\cos\varphi\dfrac{\partial T}{\partial\varphi}\right)\right\} - \dot\zeta\dfrac{\partial T}{\partial\zeta} - \dfrac{1}{\rho c_{vd}}pD + Q_T
$$

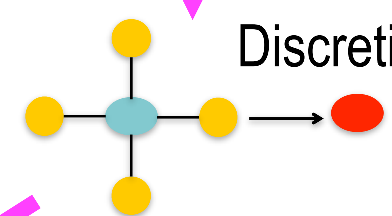
water
$$
\begin{cases}
\dfrac{\partial q^v}{\partial t} = -\left\{\dfrac{1}{a\cos\varphi}\left(u\dfrac{\partial q^v}{\partial\lambda} + v\cos\varphi\dfrac{\partial q^v}{\partial\varphi}\right)\right\} - \dot\zeta\dfrac{\partial q^v}{\partial\zeta} - (S^l + S^f) + M_{q^v} \\
\dfrac{\partial q^{l,f}}{\partial t} = -\left\{\dfrac{1}{a\cos\varphi}\left(u\dfrac{\partial q^{l,f}}{\partial\lambda} + v\cos\varphi\dfrac{\partial q^{l,f}}{\partial\varphi}\right)\right\} - \dot\zeta\dfrac{\partial q^{l,f}}{\partial\zeta} - \dfrac{g}{\sqrt{\gamma}}\dfrac{\rho_0}{\rho}\dfrac{\partial P_{l,f}}{\partial\zeta} + S^{l,f} + M_{q^{l,f}}
\end{cases}
$$

turbulence
$$
\dfrac{\partial e_t}{\partial t} = -\left\{\dfrac{1}{a\cos\varphi}\left(u\dfrac{\partial e_t}{\partial\lambda} + v\cos\varphi\dfrac{\partial e_t}{\partial\varphi}\right)\right\} - \dot\zeta\dfrac{\partial e_t}{\partial\zeta} + K_m^v\dfrac{g\rho_0}{\sqrt{\gamma}}\left\{\left(\dfrac{\partial u}{\partial\zeta}\right)^2 + \left(\dfrac{\partial v}{\partial\zeta}\right)^2\right\} + \dfrac{g}{\rho\theta_v}F^{\theta_v} - \dfrac{\sqrt{2}e_t^{3/2}}{\alpha_M l} + M_{e_t}
$$



Physical model

Mathematical description



Discretization / algorithm

Tools & Libraries

Domain science (incl. applied mathematics)

```
lap(i,j,k) = -4.0 * data(i,j,k) +
     data(i+1,j,k) + data(i-1,j,k) +
     data(i,j+1,k) + data(i,j-1,k);
```

Code / implementation

Optimal algorithm
Auto tuning

Code compilation

Computer engineering
(& computer science)

Architectural options / design

ETH zürich

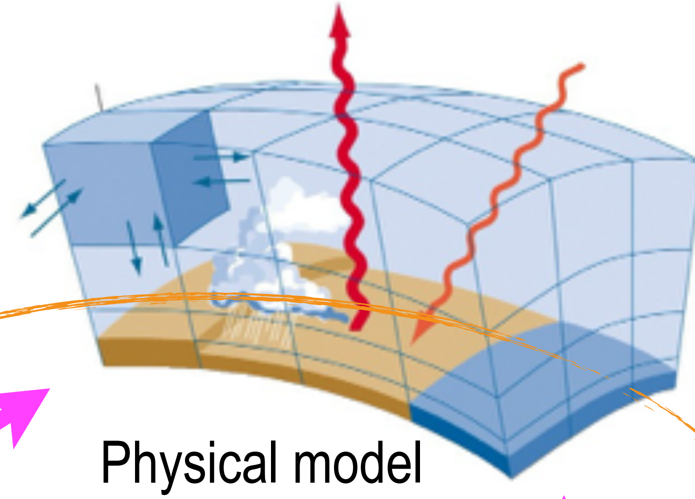
$$
\text{velocities}\begin{cases}
\dfrac{\partial u}{\partial t} = -\left\{\dfrac{1}{a\cos\varphi}\dfrac{\partial E_h}{\partial\lambda} - vV_a\right\} - \dot\zeta\dfrac{\partial u}{\partial\zeta} - \dfrac{1}{\rho a\cos\varphi}\left(\dfrac{\partial p'}{\partial\lambda} - \dfrac{1}{\sqrt{\gamma}}\dfrac{\partial p_0}{\partial\lambda}\dfrac{\partial p'}{\partial\zeta}\right) + M_u \\[2ex]
\dfrac{\partial v}{\partial t} = -\left\{\dfrac{1}{a}\dfrac{\partial E_h}{\partial\varphi} + uV_a\right\} - \dot\zeta\dfrac{\partial v}{\partial\zeta} - \dfrac{1}{\rho a}\left(\dfrac{\partial p'}{\partial\varphi} - \dfrac{1}{\sqrt{\gamma}}\dfrac{\partial p_0}{\partial\varphi}\dfrac{\partial p'}{\partial\zeta}\right) + M_v \\[2ex]
\dfrac{\partial w}{\partial t} = -\left\{\dfrac{1}{a\cos\varphi}\left(u\dfrac{\partial w}{\partial\lambda} + v\cos\varphi\dfrac{\partial w}{\partial\varphi}\right)\right\} - \dot\zeta\dfrac{\partial w}{\partial\zeta} + \dfrac{g}{\sqrt{\gamma}}\dfrac{\rho_0}{\rho}\dfrac{\partial p'}{\partial\zeta} + M_w + g\dfrac{\rho_0}{\rho}\left\{\dfrac{(T-T_0)}{T} - \dfrac{T_0 p'}{T p_0} + \left(\dfrac{R_v}{R_d} - 1\right)q^v - q^l - q^f\right\}
\end{cases}
$$

$$
\text{pressure}\quad \frac{\partial p'}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial p'}{\partial\lambda} + v\cos\varphi\frac{\partial p'}{\partial\varphi}\right)\right\} - \dot\zeta\frac{\partial p'}{\partial\zeta} + g\rho_0 w - \frac{c_{pd}}{c_{vd}}pD
$$

$$
\text{temperature}\quad \frac{\partial T}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial T}{\partial\lambda} + v\cos\varphi\frac{\partial T}{\partial\varphi}\right)\right\} - \dot\zeta\frac{\partial T}{\partial\zeta} - \frac{1}{\rho c_{vd}}pD + Q_T
$$

$$
\text{water}\begin{cases}
\dfrac{\partial q^v}{\partial t} = -\left\{\dfrac{1}{a\cos\varphi}\left(u\dfrac{\partial q^v}{\partial\lambda} + v\cos\varphi\dfrac{\partial q^v}{\partial\varphi}\right)\right\} - \dot\zeta\dfrac{\partial q^v}{\partial\zeta} - (S^l + S^f) + M_{q^v} \\[2ex]
\dfrac{\partial q^{l,f}}{\partial t} = -\left\{\dfrac{1}{a\cos\varphi}\left(u\dfrac{\partial q^{l,f}}{\partial\lambda} + v\cos\varphi\dfrac{\partial q^{l,f}}{\partial\varphi}\right)\right\} - \dot\zeta\dfrac{\partial q^{l,f}}{\partial\zeta} - \dfrac{g}{\sqrt{\gamma}}\dfrac{\rho_0}{\rho}\dfrac{\partial P_{l,f}}{\partial\zeta} + S^{l,f} + M_{q^{l,f}}
\end{cases}
$$

$$
\text{turbulence}\quad \frac{\partial e_t}{\partial t} = -\left\{\frac{1}{a\cos\varphi}\left(u\frac{\partial e_t}{\partial\lambda} + v\cos\varphi\frac{\partial e_t}{\partial\varphi}\right)\right\} - \dot\zeta\frac{\partial e_t}{\partial\zeta} + K_m^v\frac{g\rho_0}{\sqrt{\gamma}}\left\{\left(\frac{\partial u}{\partial\zeta}\right)^2 + \left(\frac{\partial v}{\partial\zeta}\right)^2\right\} + \frac{g}{\rho\theta_v}F^{\theta_v} - \frac{\sqrt{2}e_t^{3/2}}{\alpha_M l} + M_{e_t}
$$



Model development based on Python or equivalent dynamic language

Physical model

Mathematical description

Discretization / algorithm

Domain science

Tools & Libraries

```
lap(i,j,k) = -4.0 * data(i,j,k) +
     data(i+1,j,k) + data(i-1,j,k) +
     data(i,j+1,k) + data(i,j-1,k);
```

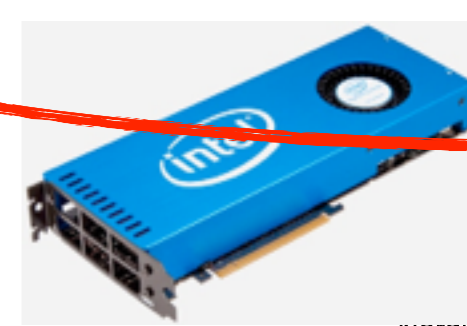
Code / implementation

Code compilation

Optimal algorithm
Auto tuning

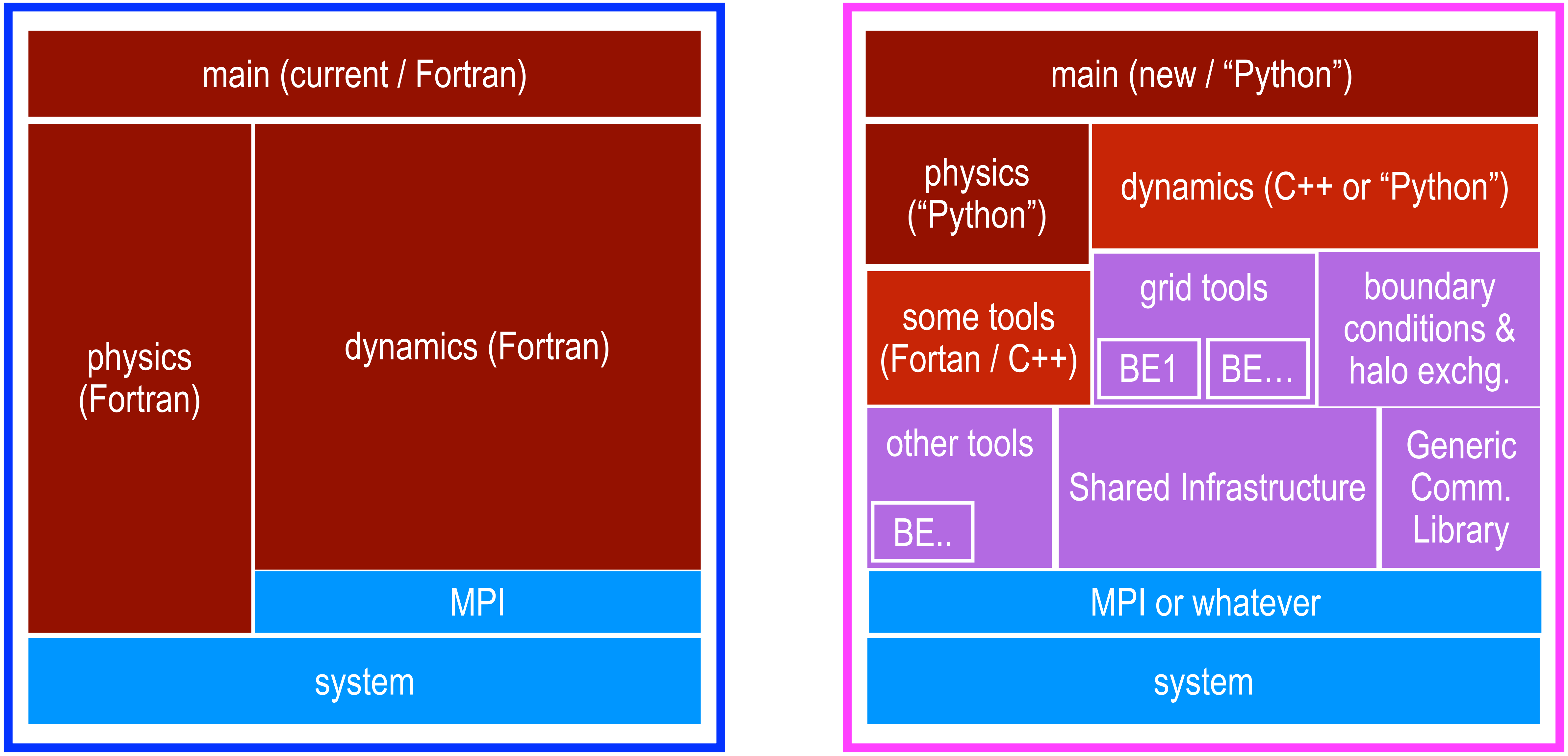
Computer engineering (& computer science)

Architectural options / design



co-design

COSMO in five year: current and new (2019) code



Thank you!