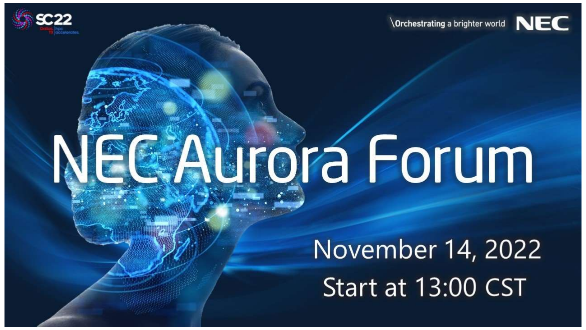


\Orchestrating a brighter world NEC

From seismic imaging to wind turbine modelling: benefits of the NEC SX-Aurora Vector Engine for the energy sector

Vincent Etienne, NEC Deutschland

NEC Aurora Forum
November 14, 2022

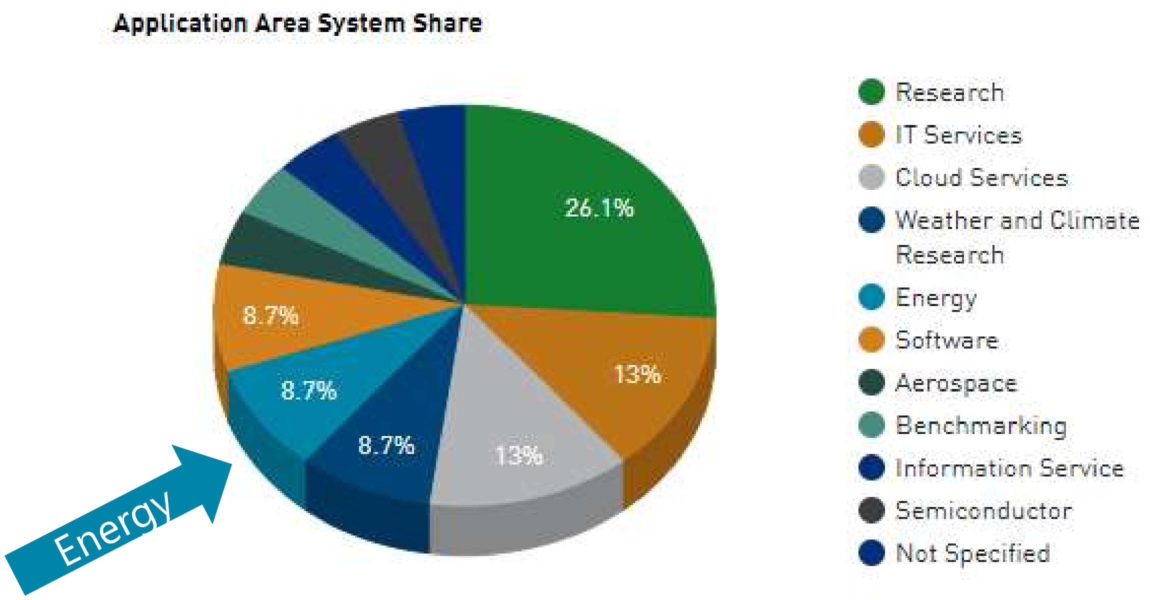
Table of content

Introduction

HPC for the O&G
New energies, new HPC workloads

Introduction – HPC for the Energy sector

June 2022



Energy is #1 with Weather Forecast for number of HPC systems
In front of Aerospace, Car manufacturing & other industries

Introduction – HPC for the O&G

History of O&G systems in Top 10

- Nov 2019 None
- Jun 2020 #6 ENI → 1st industrial system in Top 10 is O&G
- Nov 2020 #8 ENI and #10 Saudi Aramco → 1st time 2 O&G systems
- Jun 2021 #9 ENI
- Nov 2021 #9 ENI
- June 2022 None

Today (June 2022) - O&G systems in Top 100

- #12 ENI (GPU)
- #18 Saudi Aramco (GPU)
- #28 Saudi Aramco (CPU)
- #33 TotalEnergies (GPU)
- #44 ENI (GPU)
- #60 Petrobras (GPU)
- #101 TotalEnergies (CPU)

Large investment of Oil majors in HPC
Significant increase over the past few years

Introduction – New energies, new HPC workloads

To reduce carbon footprint, renewable energies are raising a lot of interests in our society
Especially in the O&G community

Introduction – New energies, new HPC workloads

Most of oil majors are investing heavily in renewables, such as wind and solar, as they look to transition towards cleaner energy sources

Home > Our Company > Our expertise > Explore and produce > Renewable energies

TotalEnergies

Strengthening our presence in renewable energies

Electricity demand is set to rise faster than global demand for energy as a whole in the coming years. According to the International Energy Agency's Sustainable Development Scenario, renewable energies will represent more than 35% of the world's energy mix in 2040. To support this growth, our ambition is to achieve 100 gigawatts of gross installed renewable power generation capacity by 2030, through the development of our solar and wind energy businesses around the world.

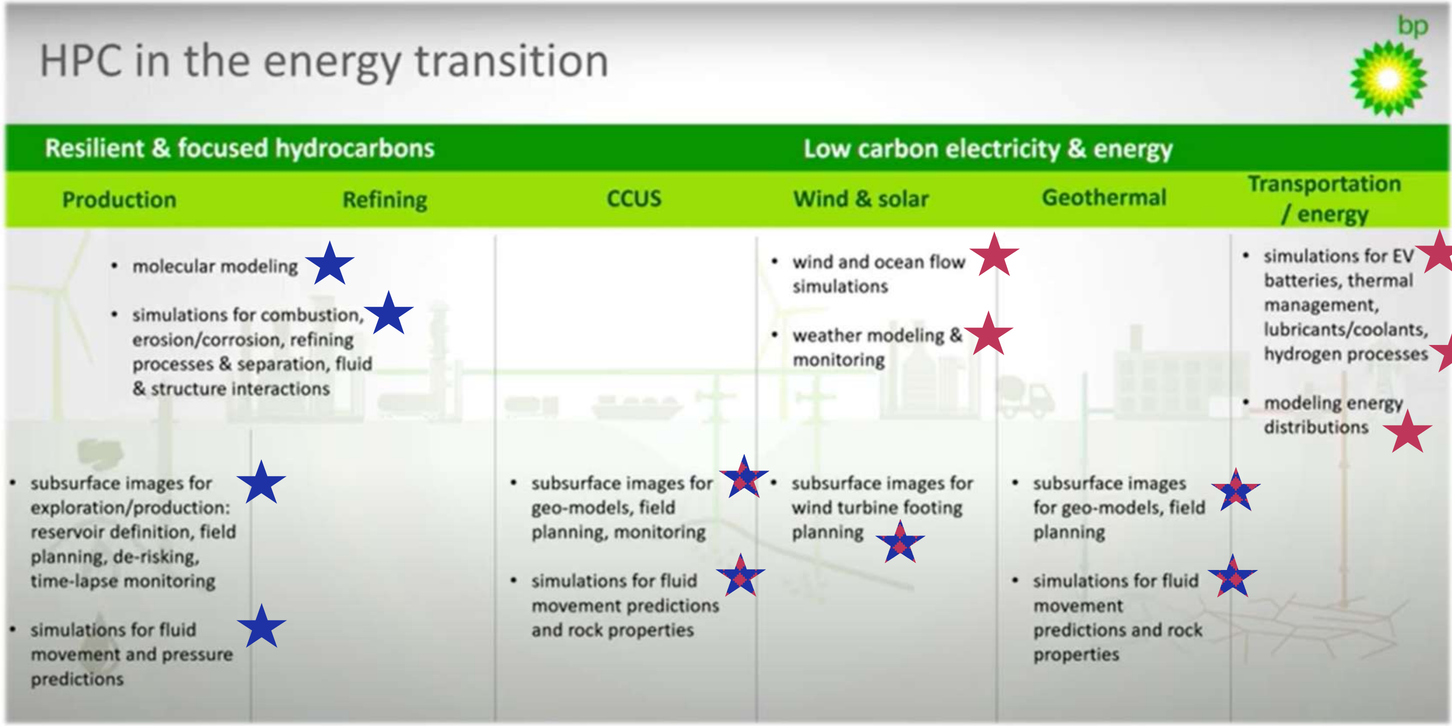
Solar and wind: Our ambition in renewable energies

Confident of the opportunities afforded by renewable energies, we aim to attain electricity production of 120 TWh by 2030, mainly through the development of solar and wind power.

From TotalEnergies website

Introduction – New energies, new HPC workloads

Besides traditional applications for O&G, new ones are expected to increase the HPC requirements



Source web (adapted): HPC and ML Aspects of the Energy Transition (RICE university)

★ Traditional application ★ Trad. with new use case ★ New application

What are the compute footprints of traditional applications for O&G?

How different it will be for new kind of applications involved in the energy transition?

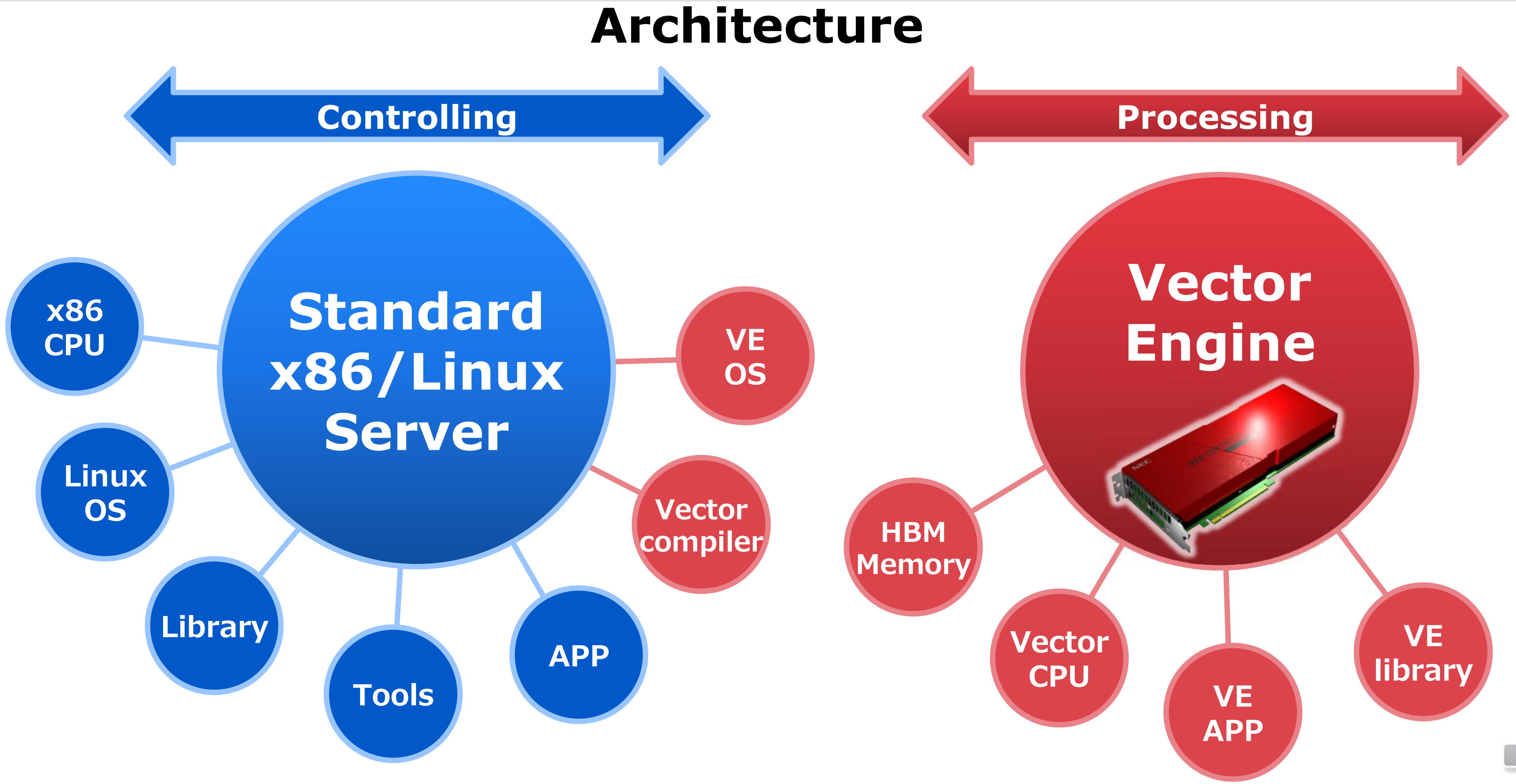
What are the benefits of NEC Vector Engine?

SX-Aurora TSUBASA Vector Engine

Overview

Native & accelerator modes

NEC SX-Aurora TSUBASA Vector Engine - Overview



NEC SX-Aurora TSUBASA Vector Engine - Overview

VE20 (current generation)
Vector length 256 double floats (512 single)
64 vector registers

Processor Version	Type A	Type B
Cores/processor	10	8
Core performance	307GF (DP) 614GF (SP)	
Processor performance	3.07TF (DP) 6.14TF (SP)	2.45TF (DP) 4.91TF (SP)
Cache capacity	16MB	
Cache bandwidth	3TB/s	
Cache Function	Software Controllable	
Memory capacity	48GB	
Memory bandwidth	1.53TB/s	
Power	~300W (TDP) ~200W (Application)	

Development Philosophy

POINT 1 Strong cores

Highest memory bandwidth per core
Highest peak performance per core

POINT 2 Easy to Use

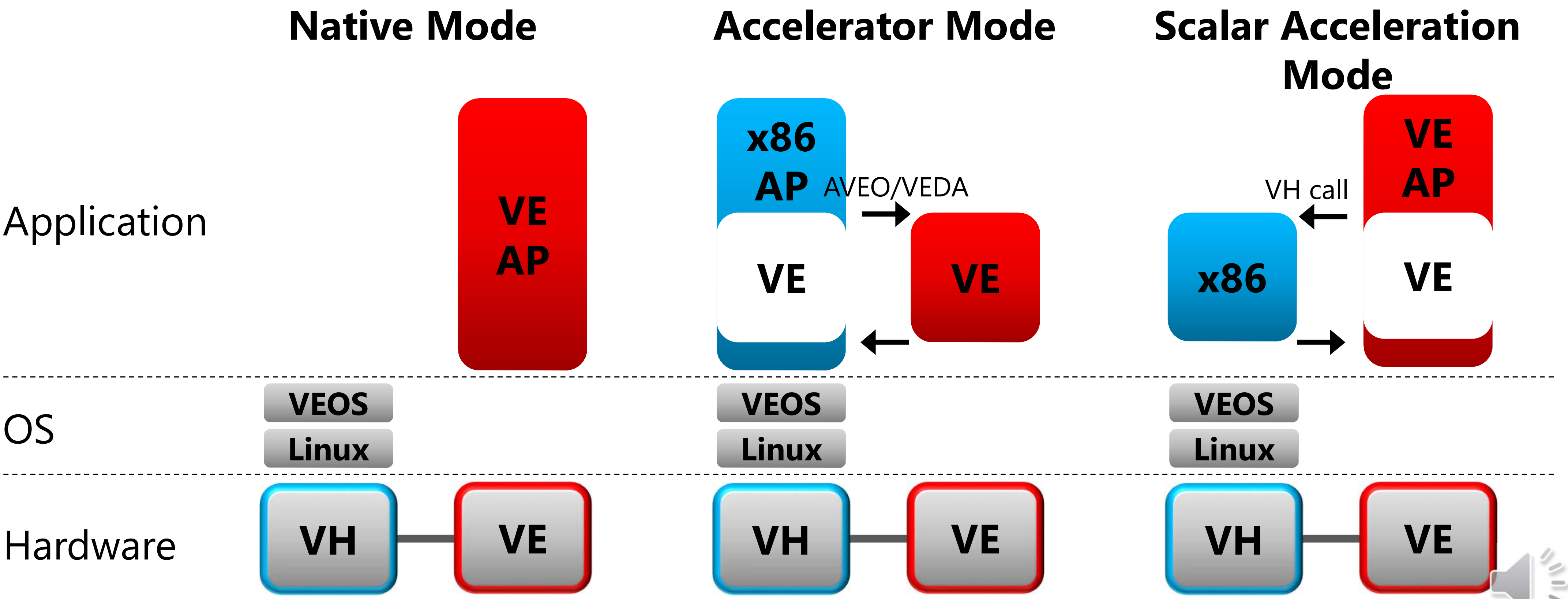
Fortran/C/C++ programming, OpenMP
Automatic vectorization/parallelization

POINT 3 Energy Efficient

Power limitation for HPC systems becomes an issue
Higher sustained performance with lower power

NEC SX-Aurora TSUBASA VE – Native & accelerator modes

Vector Engine (VE) programming model: **a unique feature**



  NEC Group Public Use \Orchestrating a brighter world NEC

NEC SX-Aurora TSUBASA VE – Native & accelerator modes

Example with OpenFAST

- Open-source wind turbine simulation tool
  - Fortran
  - OpenMP
- Dependencies
  - BLAS
  - LAPACK
- Compilation with Cmake

Porting on SX-Aurora with Native mode

- Straight forward compilation with nfort
  - Natural support for Fortran and OpenMP
- Linking with NLC (NEC Numeric Library Collection)

NEC NLC

Library		Functions
ASL	Native Interface	Scientific library with a wide variety of algorithms for numerical/statistical calculations
	Unified Interface	Fourier Transforms, Random Number Generators, Sortings
	FFTW3 Interface	Interface library to use Fourier Transform functions of ASL with FFTW (version 3.x) API
BLAS		Basic Linear Algebra Subprograms
LAPACK		Linear Algebra PACKage Simultaneous linear equations, Eigenvalue equations, and Singular value decomposition
ScaLAPACK		Scalable Linear Algebra PACKage Simultaneous linear equations, Eigenvalue equations, and Singular value decomposition (for distributed memory parallel programs)
BLACS		Basic Linear Algebra Communication Subprograms (uses MPI) Message passing library for performing basic operations on vectors and matrices (for distributed memory parallel programs)
SBLAS		Sparse BLAS (from ACM Algorithm 692) Basic operations of sparse matrices
HeteroSolver		Simultaneous linear equations (Direct sparse solver)
Stencil Code Accelerator		Stencil Code Acceleration

NEC SX-Aurora TSUBASA VE – Native & accelerator modes

Example with OpenFAST

- Open-source wind turbine simulation tool
  - Fortran
  - OpenMP
- Dependencies
  - BLAS
  - LAPACK
- Compilation with Cmake

Porting on SX-Aurora with Native mode

- Straight forward compilation with nfort
  - Natural support for Fortran and OpenMP
- Linking with NLC (NEC Numeric Library Collection)

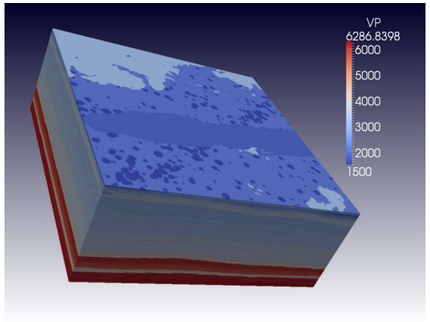
Testing / benchmarking on SX-Aurora in progress

```
[vetienne@arrproto27 Release_nec_openmp]$ ./glue-codes/openfast/openfast -v

 **************************************************************************************************
 OpenFAST

 Copyright (C) 2022 National Renewable Energy Laboratory
 Copyright (C) 2022 Envision Energy USA LTD

 This program is licensed under Apache License Version 2.0 and comes with ABSOLUTELY NO WARRANTY.
 See the "LICENSE" file distributed with this software for details.
 **************************************************************************************************

 OpenFAST-v3.2.1-dirty
 Compile Info:
  - Compiler: nfort (NFORT) 3.5.1
  - Architecture: 64 bit
  - Precision: double
  - OpenMP: Yes, number of threads: 8/8
  - Date: Sep 13 2022
  - Time: 10:13:33
 Execution Info:
  - Date: 09/13/2022
  - Time: 11:11:49+0200


  OpenFAST terminated normally.
```


NEC SX-Aurora TSUBASA VE – Native & accelerator modes

Example of a seismic imaging application
Porting with accelerator mode

Read model on disk
n1 x n2 x n3



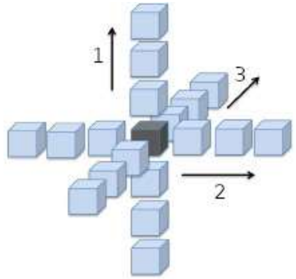
Lots of disk IOs & data manipulations
→ Better do it on CPU (scalar operations)

Locate source and receivers positions in grid

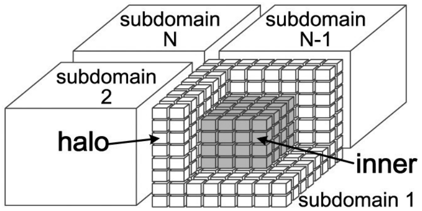
Loop on time steps Nt

Compute with finite difference

Computation FD kernels localized
→ **Offloading kernels** on SX-Aurora appropriate

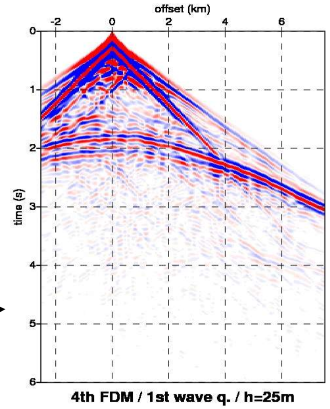


Exchange halos



Record traces

Write traces



NEC SX-Aurora TSUBASA VE – Native & accelerator modes

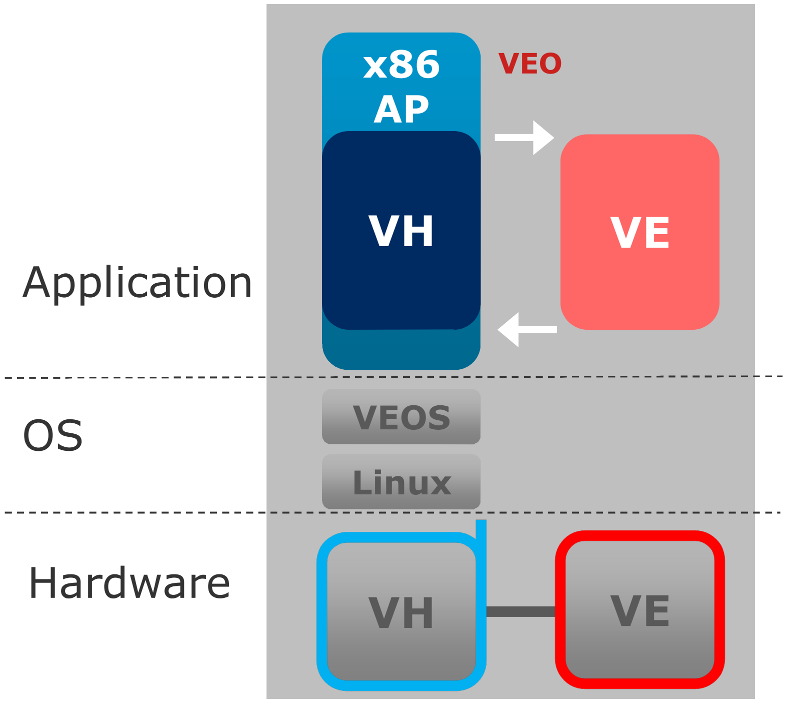
Accelerator mode with VEDA

- Offloading library developed by NEC LAB Europe
- Implements CUDA-like APIs
- Support C++

Benefits

- When existing code supports both CPU & GPU
  - Offloading sections already identified
  - Could **reuse CPU kernels** as is in VEDA kernels
- **Porting with VEDA is easy**
  - No need to learn new programing model → **Great benefit compared to constructor specific (ex: CUDA)**

Offloading with VEDA



NEC SX-Aurora TSUBASA VE – Native & accelerator modes

Original kernel CPU version

- y = ax + b
- C++
- simple OpenMP construct

```
    template<typename ValueType>
    void eq_axpb(ValueType* dst, ValueType a,
                 const ValueType* x, ValueType b, size_t n) {
#pragma omp parallel for schedule(static) default(none) shared(dst,n,x,a,b)
        for (size_t i=0;i<n;i++) dst[i]=a*x[i]+b;
    }
```

compiled with g++ or icpc

NEC SX-Aurora TSUBASA VE – Native & accelerator modes

Original kernel CPU version

- y = ax + b
- C++
- simple OpenMP construct

```
template<typename ValueType>
void eq_axpb(ValueType* dst, ValueType a,
             const ValueType* x, ValueType b, size_t n) {
#pragma omp parallel for schedule(static) default(none) shared(dst,n,x,a,b)
    for (size_t i=0;i<n;i++) dst[i]=a*x[i]+b;
}
```

compiled with g++ or icpc

Equivalent VEDA version

- Launch VEDA kernel from CPU
- Similar to a CUDA call

```
template <typename ValueType>
void eq_axpb(ValueType *dst, ValueType a,
             const ValueType *x, ValueType b, size_t n, const geodrive::runtime::queue_t *qid)
{
    VEDAptr<ValueType> vdst = (VEDAdeviceptr)dst;
    VEDAptr<ValueType> vx = (VEDAdeviceptr)x;
    VEDAfunction func = Ctxt::vedaGetFunctionFromLibrary<ValueType>(Ctxt::vedaGetSystemLibrary(), "eq_axpb");
    VEDACHK(vedaLaunchKernel(func, 0, vdst.ptr(), a, vx.ptr(), b, n), "vedaLaunchKernel failed");
    Ctxt::sync();
}
```

compiled with g++ or icpc

NEC SX-Aurora TSUBASA VE – Native & accelerator modes

kernel VEDA version

- y = ax + b
- C++
- simple OpenMP construct

```
extern "C" void VEDA_FNAME(eq_axpb, ValueType) (ValueType *dst, ValueType a,
                                                const ValueType *x, ValueType b, size_t n)
{
#pragma omp parallel for schedule(static) default(none) shared(dst, n, x, a, b)
    for (size_t i = 0; i < n; i++)
        dst[i] = a * x[i] + b;
}
```

compiled with nc++
C++ original CPU kernel as is

Equivalent VEDA version

- Launch VEDA kernel from CPU
- Similar to a CUDA call

```
template <typename ValueType>
void eq_axpb(ValueType *dst, ValueType a,
             const ValueType *x, ValueType b, size_t n, const geodrive::runtime::queue_t *qid)
{
    VEDAptr<ValueType> vdst = (VEDAdeviceptr)dst;
    VEDAptr<ValueType> vx = (VEDAdeviceptr)x;
    VEDAfunction func = Ctxt::vedaGetFunctionFromLibrary<ValueType>(Ctxt::vedaGetSystemLibrary(), "eq_axpb");
    VEDACHK(vedaLaunchKernel(func, 0, vdst.ptr(), a, vx.ptr(), b, n), "vedaLaunchKernel failed");
    Ctxt::sync();
}
```

compiled with g++ or icpc

Traditional applications for O&G

Seismic imaging & full waveform inversion

Reservoir simulation

Geosciences & NEC
A bit of history with SPECFEM3D (Seismic/earthquake modelling)


SC2003 IGNITING INNOVATION

SC2003 CONFERENCE PHOENIX, AZ NOVEMBER 15-21, 2003

SC2003 Gordon Bell Award

Dimitri Komatitsch

California Institute of Technology

A 14.6 Billion Degrees of Freedom, 5 Teraflops, 2.5 Terabyte Earthquake Simulation on the Earth Simulator

SC2003 GENERAL CHAIR

SC2003 TECHNICAL PROGRAM CO-CHAIR

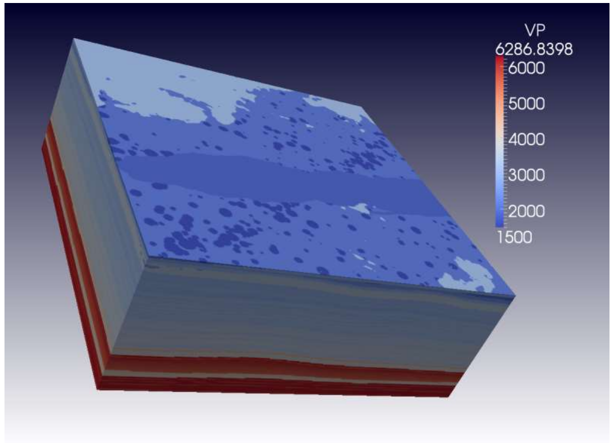
جامعة الملك عبدالله للعلوم والتقنية

King Abdullah University of Science and Technology

Collaboration with D. Peter
Main developer of SPECFEM3D

Traditional applications for O&G

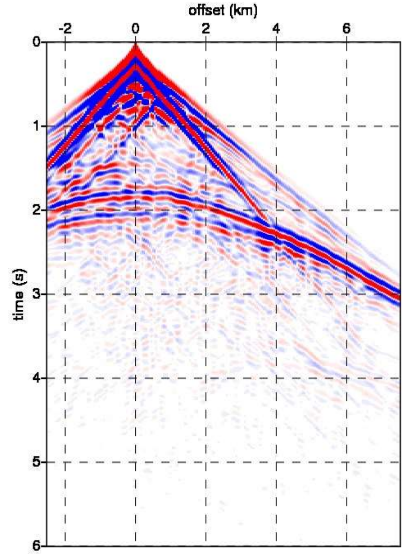
In O&G
3 main applications
Reservoir simulation
Seismic imaging & inversion



Model

Seismic modeling
Forward problem

Seismic inversion
Inverse problem



Data

Compute intensive applications that require HPC systems

Ex. Elastic modeling in large scale survey about 100 EFlop

HPC

Traditional applications – Memory bandwidth

Representative products available on the market (2021)

Spec.	CPU	GPU	VE
# cores	2 x 38	5 120	8
Memory type	DDR4	HBM2	HBM2
Capacity GB	256	32	48
Peak GB/s	410	900	1351

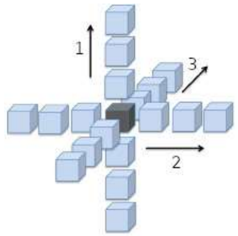
Measured memory bandwidth for simple operations on arrays

Case	CPU		GPU		VE	
	GB/s	% peak	GB/s	% peak	GB/s	% peak
Fill	200	49	725	81	1060	78
Copy	300	73	616	68	1085	80
Add	301	73	694	77	1083	80
Mul	300	73	695	77	1084	80
AddUpd	294	72	691	77	1083	80

Architectures reach 70-80 % of their peak bandwidth

Traditional applications – Seismic imaging

Main compute kernel
Finite difference operators

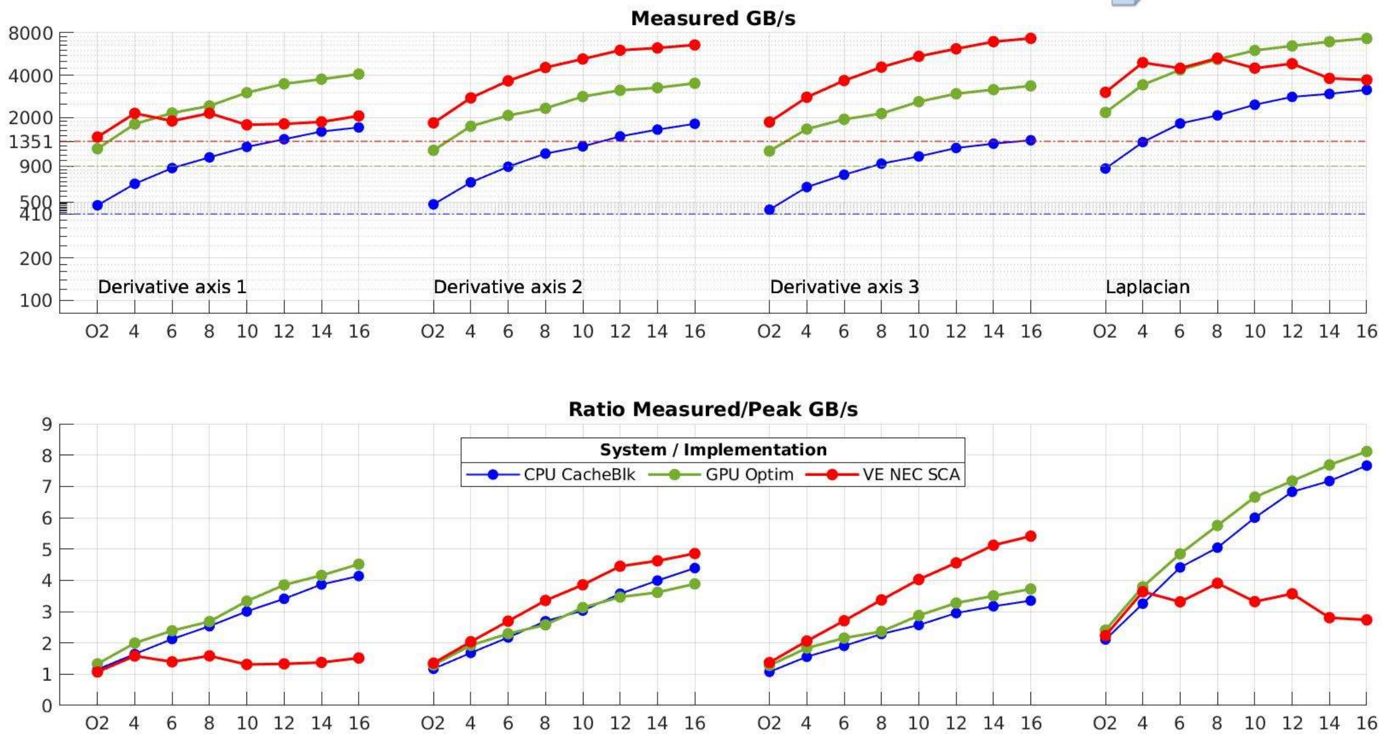


Optimal implementation for each architecture

CPU: Cache blocking

GPU: Data prefetch in local (shared) memory

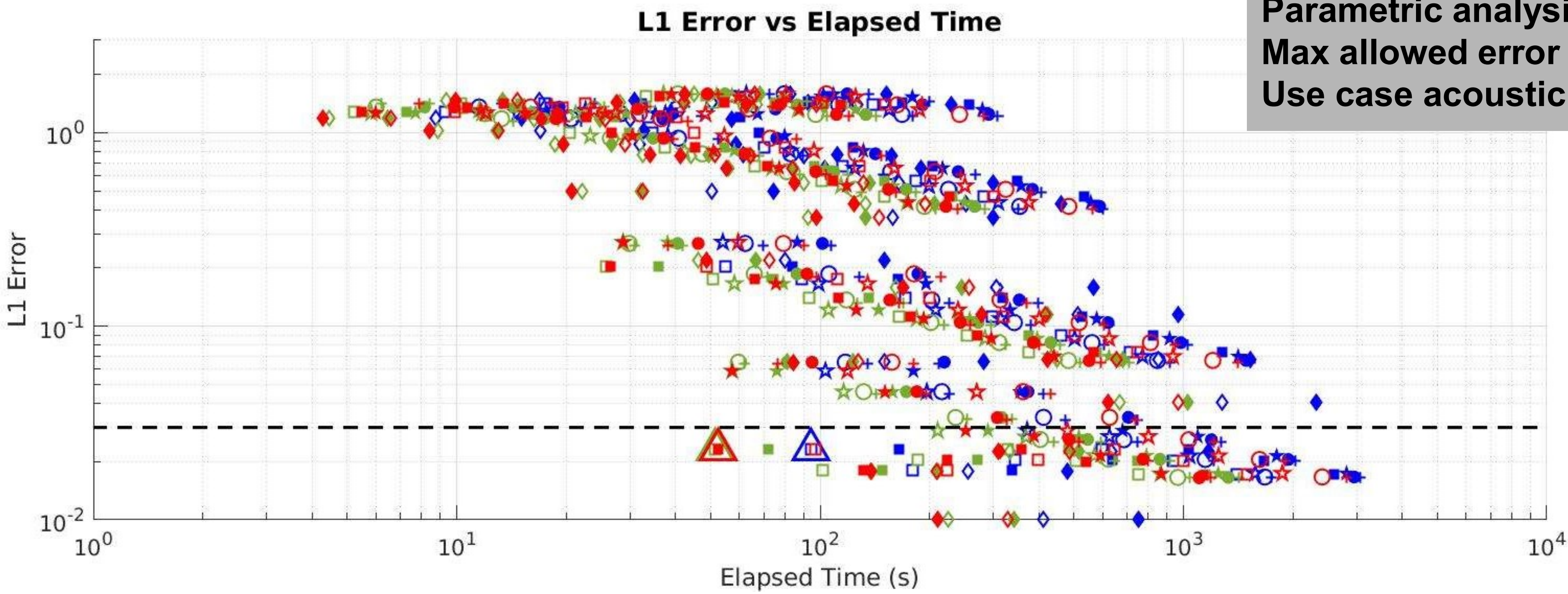
VE: NEC SCA library



EAGE HPC 2021 workshop
Benchmark tool **hpcscan**
available on GitHub

Each architecture has its specific performance 'signature'
How to select the optimum?

Traditional applications – Seismic imaging

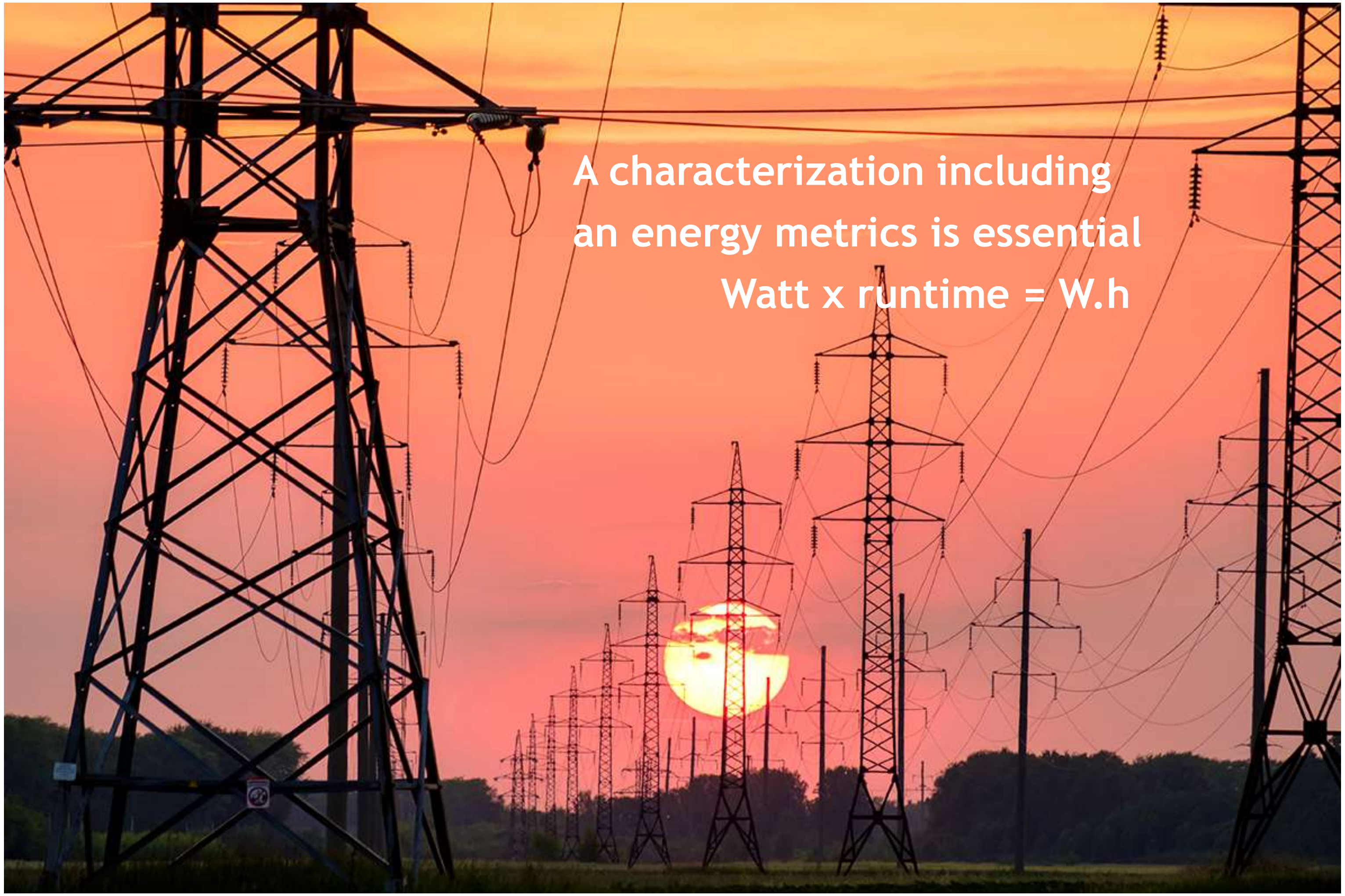


Parametric analysis
Max allowed error 3%
Use case acoustic RTM

	CPU	GPU	VE
Optimal algo.	Stand. O6	Stand. O6	Split O6
Optimal time (s)	94.4	51.4	52.5
Speed-up	1	1.8	1.8

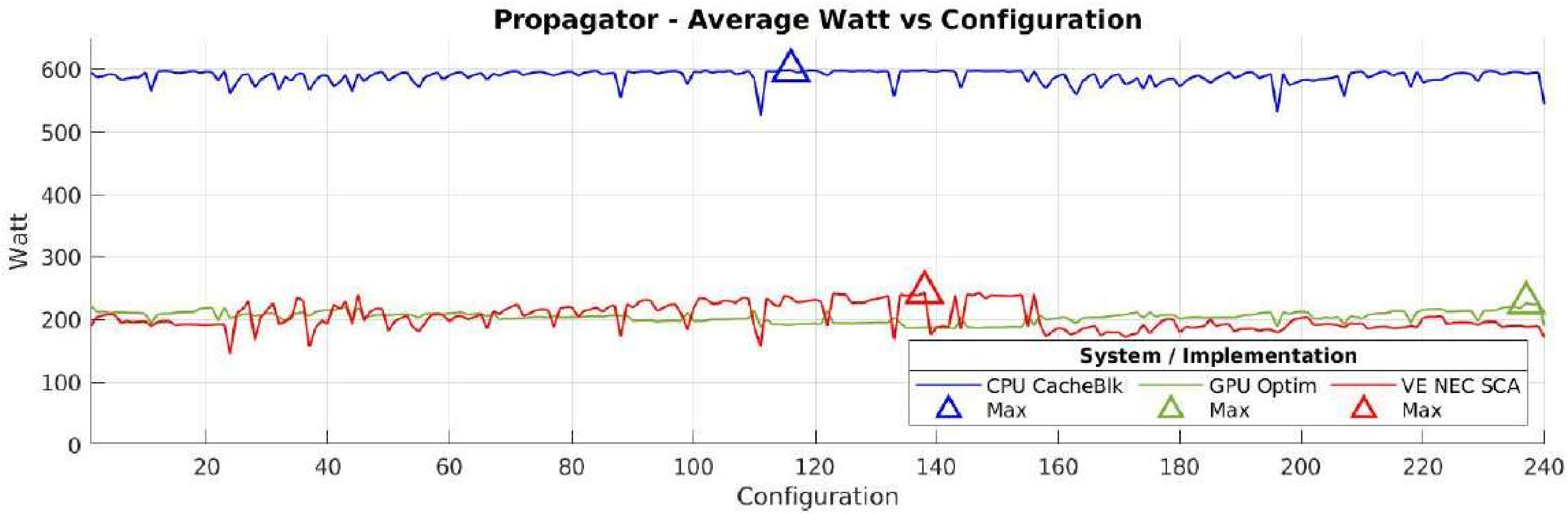
Optimal FD order 6
GPU & VE: 2x speedup vs CPU

Traditional applications – Seismic imaging



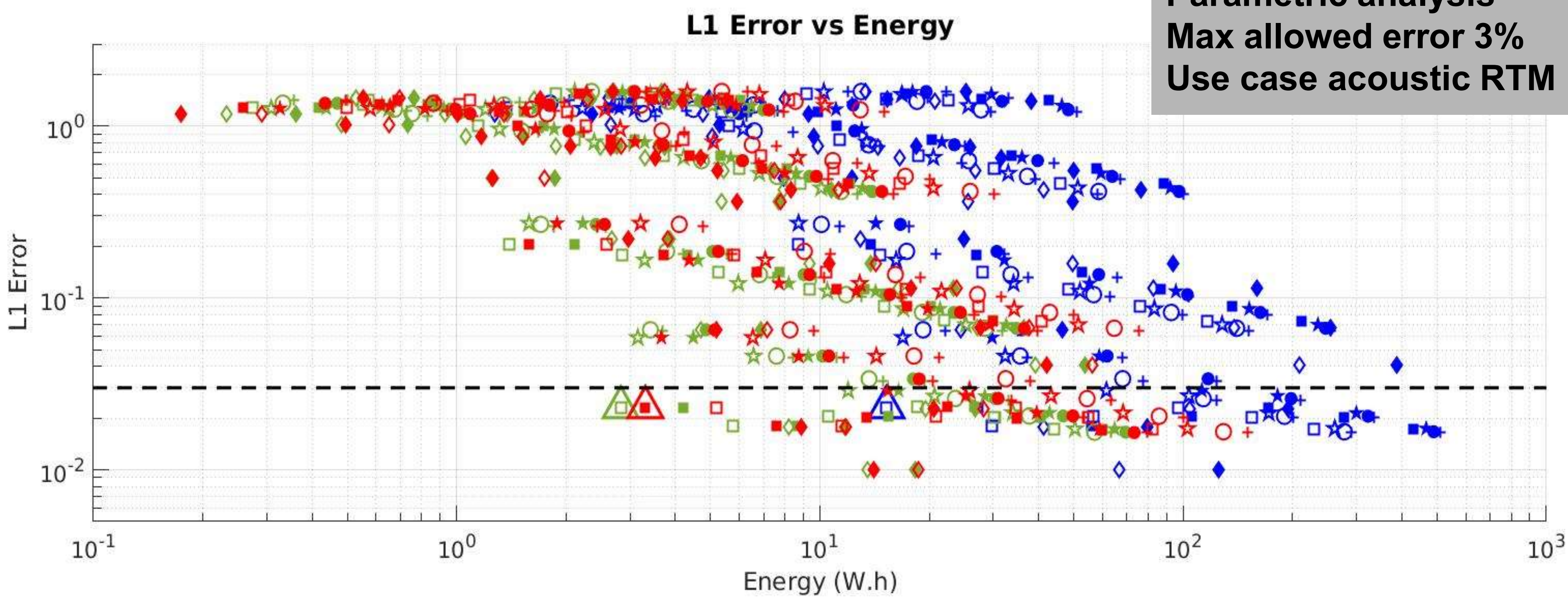
Traditional applications – Seismic imaging

Power consumption for seismic FD propagator
→ **VE & GPU about 3X less Watts than CPU**



Traditional applications – Seismic imaging

Parametric analysis
Max allowed error 3%
Use case acoustic RTM



	CPU	GPU	VE
Optimal algo.	Stand. O6 N=500 0.1 CFL	Stand. O6 N=500 0.1 CFL	Split O6 N=500 0.1 CFL
Watt.Hour	15.24	2.83	3.31
Efficiency	1	5.4	4.6

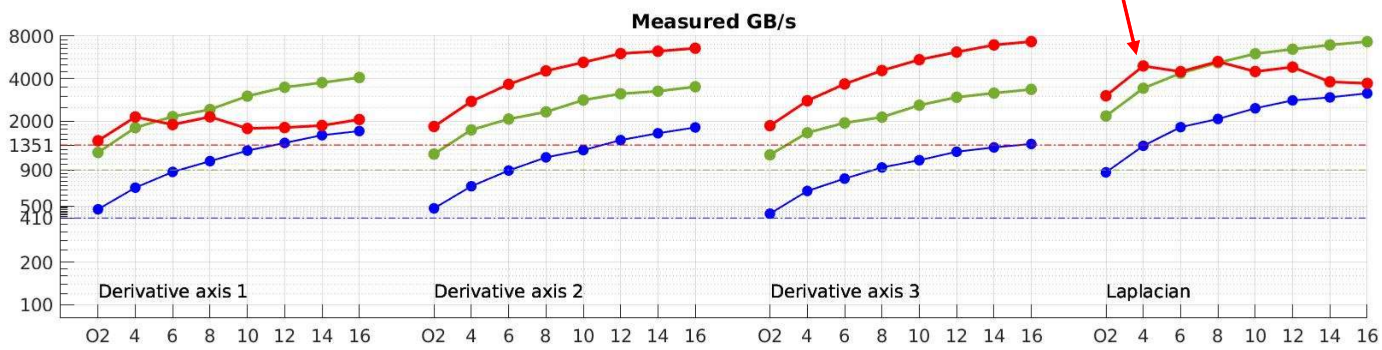
GPU & VE: Efficiency
Combines Runtime + Energy
About x5 vs CPU

Traditional applications – Full waveform inversion

FWI (seismic inversion) shares common features with RTM (seismic imaging)

But discretizations are different
- RTM (high freq.) → **coarse grids** & high order stencils
- FWI (low freq.) → **dense grids** & low order stencils

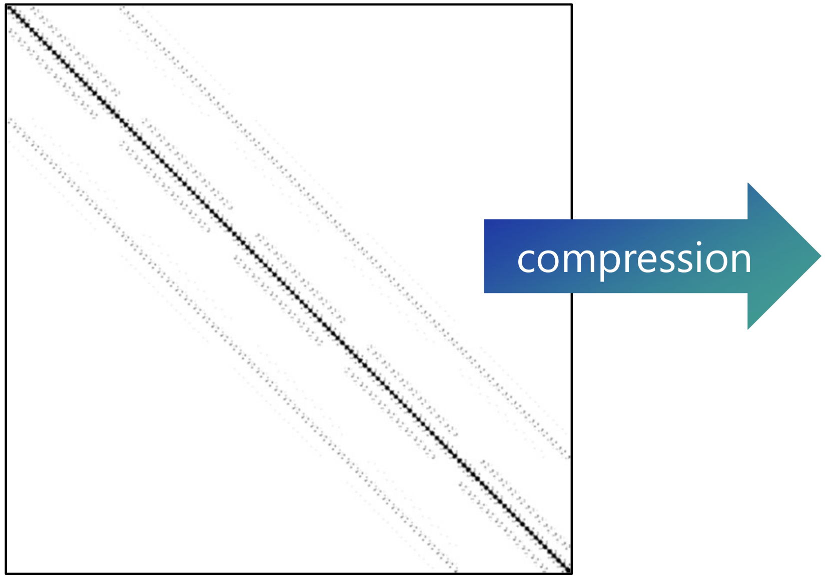
➡ Optimal FWI resolution 4 points / λ → suitable for FD O4 (VE performs best)



Traditional applications – Full waveform inversion

FWI can be formulated in the frequency domain

- Success of hierarchical approach from low to higher frequencies
- Mitigate local minimum attraction at vicinity of initial model
- In freq. domain, **direct (LU factorization) or iterative solvers can be used**



Example of acoustic impedance matrix

Collaborations with NEC

MUMPS Solver
mumps-consortium.org

Block Low-Rank approximations to improve multifrontal sparse solvers with MUMPS consortium

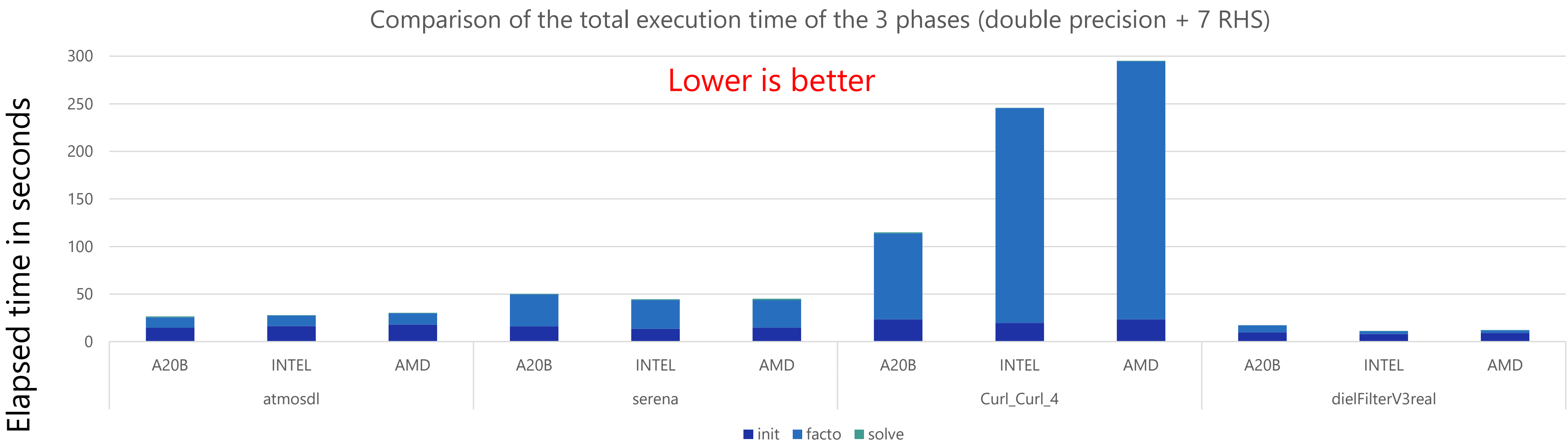
KAUST
King Abdullah University of Science and Technology

Tile Low-Rank approximations with KAUST collaboration

MUMPS Benchmark

Solvers are also the main ingredients of reservoir simulator

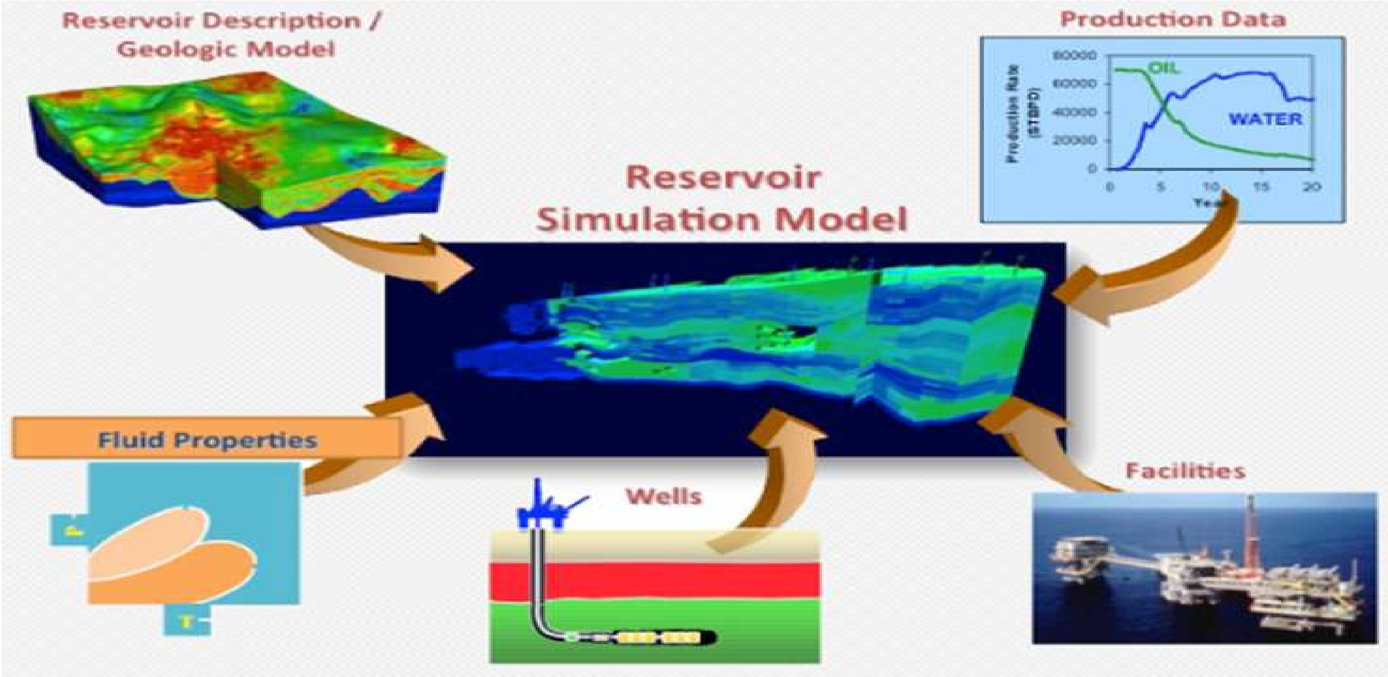
- Direct solver for sparse linear systems
  - **Aurora VE version available in current version of MUMPS**



1 VE AURORA 20B same performance as a server with 2 processors
And with much less power consumption!

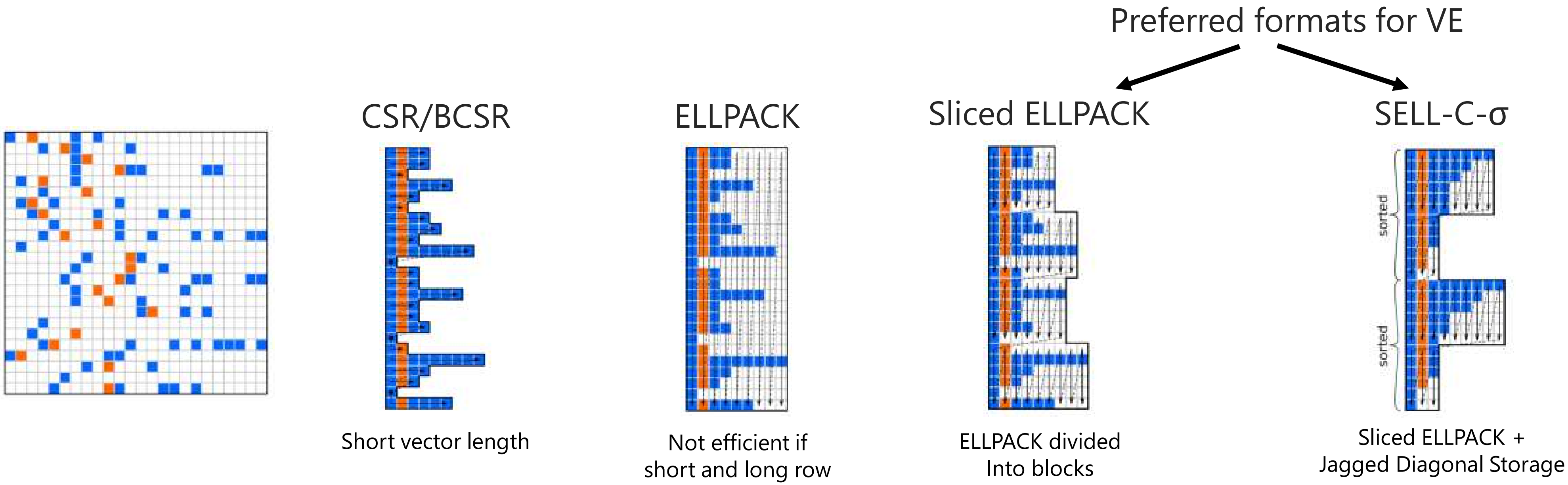
Traditional applications – Reservoir simulation

- Large models
  - Millions of cells
  - Large number of simulated years of production
- HPC KPIs
  - Memory intensive application
  - High pressure on the interconnect



Traditional applications – Reservoir simulation

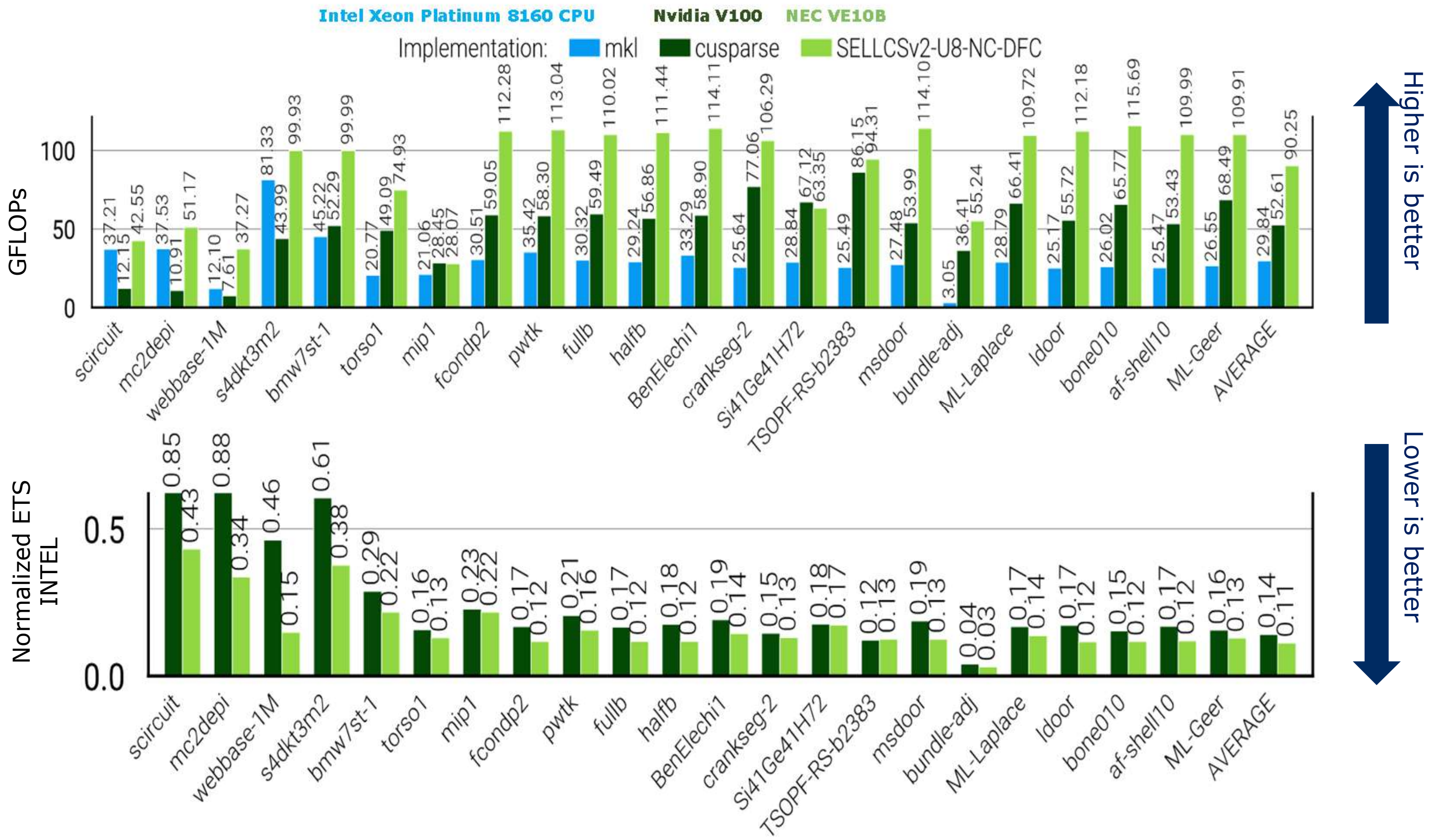
- Use sparse matrix as data structure
- **Main compute kernel is Sparse matrix-vector multiplication** (SpMV) operation
- SpMV performances depend on sparse matrix format and target architecture



Sparse Matrix Data Structures for High Performance Computing - https://faculty.cc.gatech.edu/~echow/ipcc/hpc-course/sparsemat.pdf

Traditional applications – Reservoir simulation

- Barcelona Supercomputer Center [BSC] technical report on SpMV performance comparison
- GFLOPS and Energy-to-Solution (ETS) shall be read simultaneously per each dataset
- **NEC always outperforms both Nvidia and Intel performance**
  - NEC VE's ETS is lower or in worst cases equals to Nvidia



Gómez, Constantino, M. Casas, F. Mantovani, and E. Focht. *Optimizing sparse matrix-vector multiplication in NEC SX-Aurora Vector Engine*. Technical Report, Barcelona Supercomputing Center, August 2020

New applications for O&G

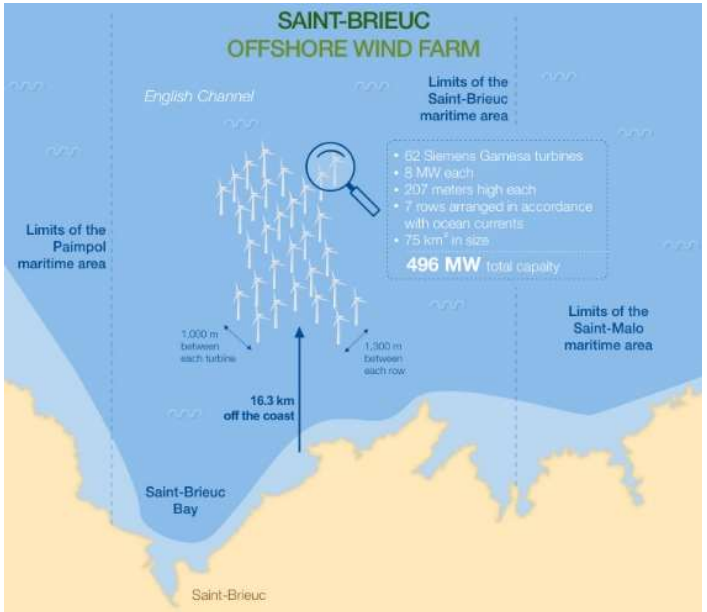
Wind turbine modeling

Carbone Capture and Storage

Optimization problems with quantum annealing

New applications – Wind turbine modeling

- According to International Energy Agency, **share of renewables** in global electricity production was **29% in 2020**
- **2/3 hydroelectricity,** and the rest **wind and solar energies**
- Considering **rapid growth of wind energy**, this resource is attracting significant interest from oil majors
- Oil companies have **experience with offshore oil rigs**. This is crucial to expedite development of offshore wind energy



New applications – Wind turbine modeling

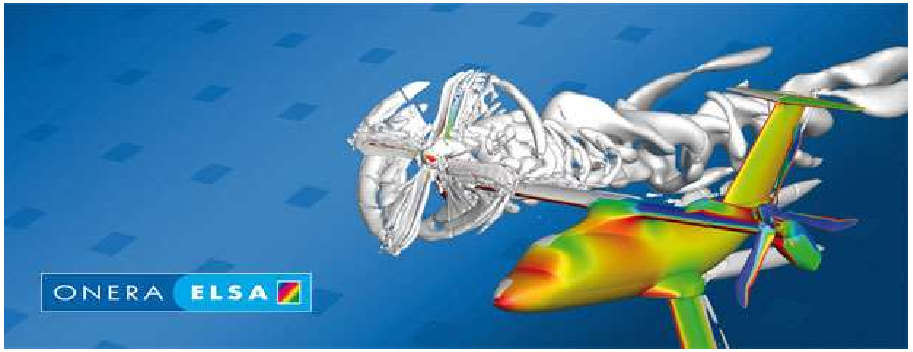
- Has common roots with reservoir simulation as it involves **computational fluid dynamics (CFD)**
- Physical phenomena are distinct
  - In reservoirs, physics governed by flow of multiphase fluids in porous rocks
  - For wind turbine, **laws of aerodynamics** to compute flow of air (compressible fluid) around rotor blades of the turbine
- Complex problem when different interactions are taken into account, as for **floating offshore wind turbines**

New applications – Wind turbine modeling

- Several numerical schemes can be used to address this challenge

  - 1st, **Blade Element Momentum Theory (BEMT)** used for long time in aerodynamics.
  Reduces 3D problem into stationary Navier-Stokes 2D equations based on approximations
  Ex: OpenFAST from US National Renewable Energy Laboratory (NREL)

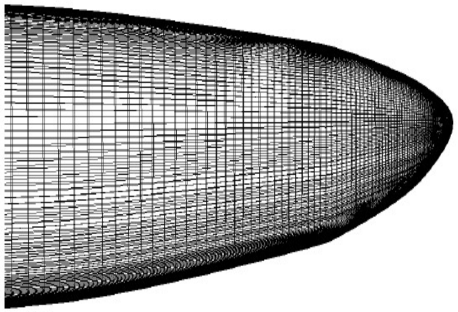
  - 2nd, **Reynolds Averaged Navier-Stokes (RANS)**
  Solves 3D problem, hence more accurate but with higher computational cost
  Ex: CFD package elsA from ONERA (French aerospace agency, former user of NEC SX systems)

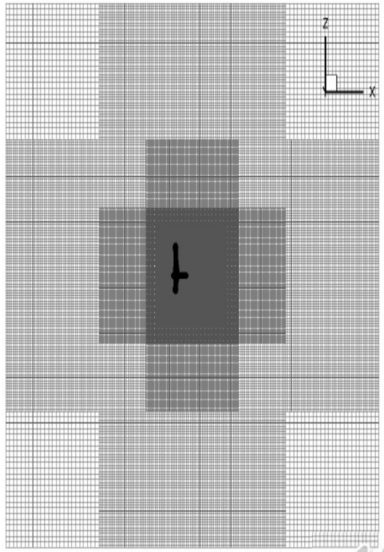


New applications – Wind turbine modeling

- elsA solves compressible RANS equations with **finite-volume cell-centered formulation** and **implicit time scheme**
- Mesh built with Chimera approach: curvilinear body-fitted grid built around rotor blades, embedded into set of Cartesian background grids
- From HPC point of view, elsA, and most CFD codes, **combine specificities of traditional O&G applications**

  - Finite-volume (i.e. modified 2nd-order FD) → **stencil-like operations** as in seismic imaging
  - Implicit time scheme → resolution of sparse linear system and **algebraic solvers** as in reservoir
  - **Expected to perform efficiently on VE** as it outperforms other architectures for low order stencils

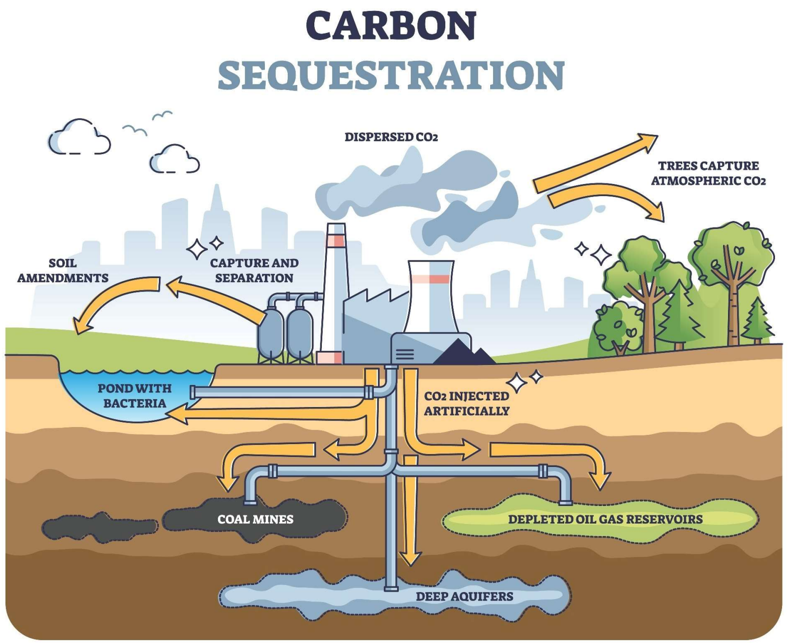




elsA mesh (from Lienard et al., 2020)

New applications – Carbone Capture & Storage

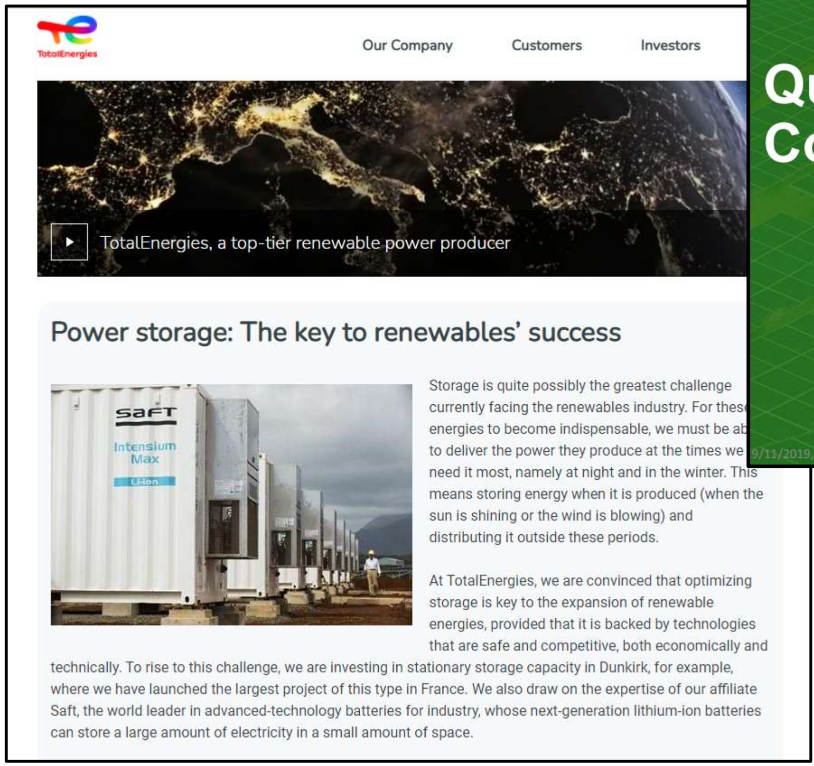
- Another **use of CFD** for **CCS modelling**
- Capturing CO2 before it enters atmosphere, transporting & storing for very long time in deep geological formations
  - Compatible with large O&G production infrastructures in place
  - CO2 injected for decades for various purposes, including enhanced oil recovery
  - But **long-term storage of CO2 is new concept and requires further developments**

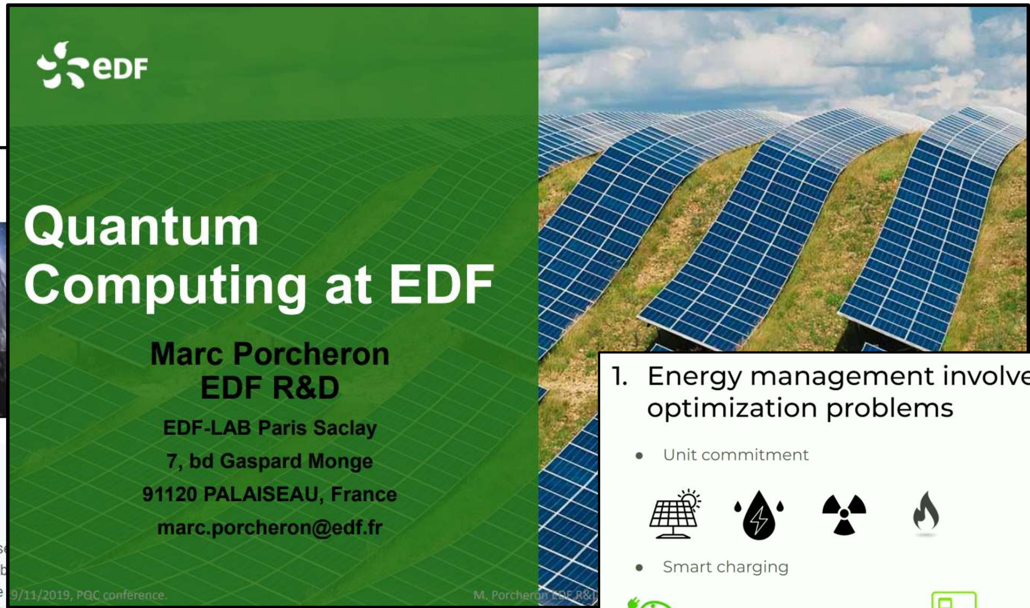


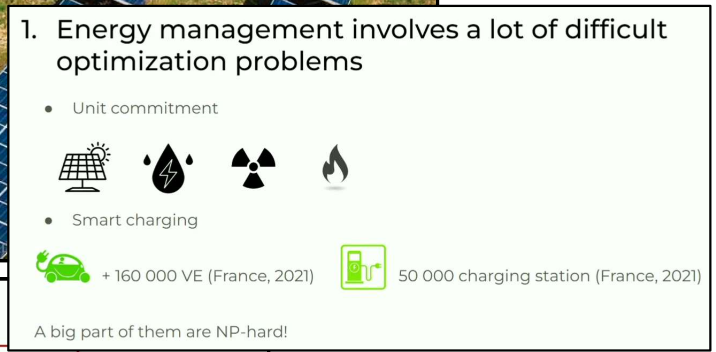
- Numerical methods for CCS modelling **similar to reservoir simulation** but with specificities
  - Complex fluid flow, thermal, and geomechanical effects, as implemented in GEOSX (Gross, 2021)
  - **We do not expect major different compute footprint** than traditional reservoir simulation

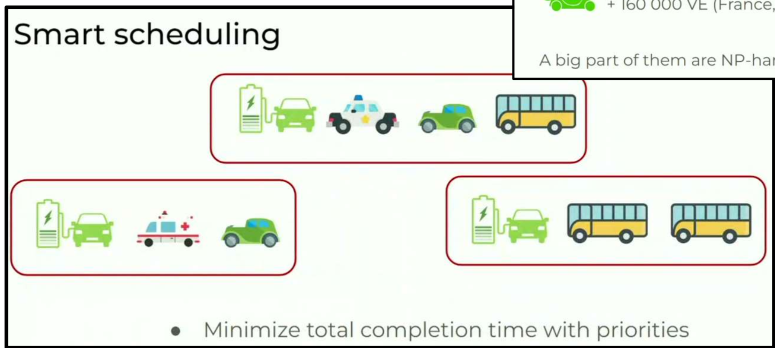
New applications – Optimization problems with quantum annealing

A wide range of applications for the energy distribution





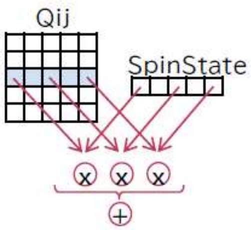




New applications – Optimization problems with quantum annealing

VA Performance is provided by:
Matrix operation acceleration by VE, large and fast memory, and optimized algorithm for VE

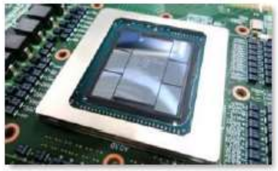
Vector operation on VE
Energy calculation is matrix operation




```
for(i=0; i<numSpins; i++)
  DeltaEnergy +=
      Qij[FlipSpin][i] x SpinState[i]
```

Effect from neighbor spin — 0 or 1

Full connect 100k qbits/VE and high memory bandwidth

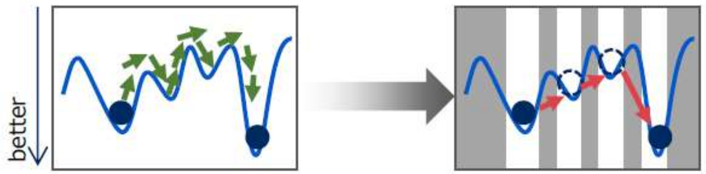


- 48GB memory capacity and 1.5TB/s memory bandwidth
- Multi card supports larger number of qbits (100k qbits x n)$^{1/2}$

Avoiding Redundant Search and Optimized algorithm for VE

Existing search — Including constraint violations

VA search — skip constraint violations



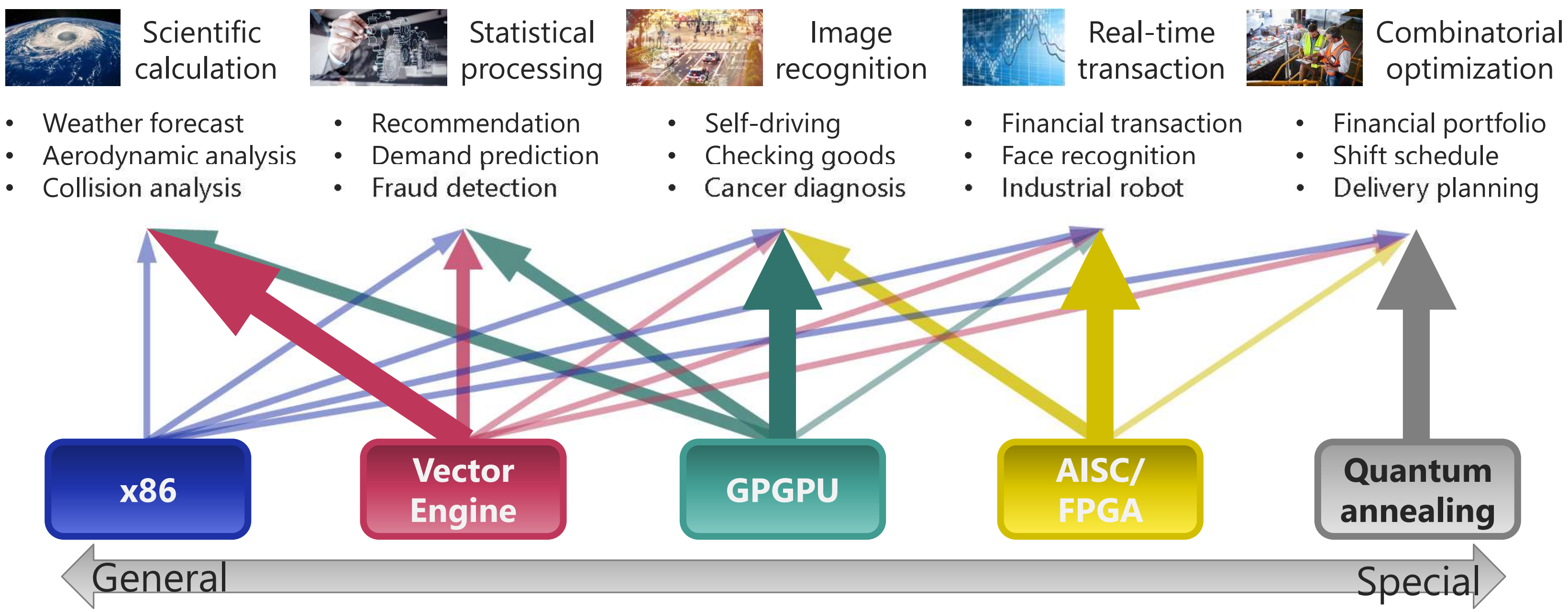
computational complexity reduction

cf. talk from Deepak Pathania presented at the NEC Aurora Forum
Application of VA for O&G

Conclusions

Conclusions

- The energy sector has large needs for HPC systems
- This demand will increase for the energy transition
  - Variety of algorithms will increase, **heterogenous computing** is expected

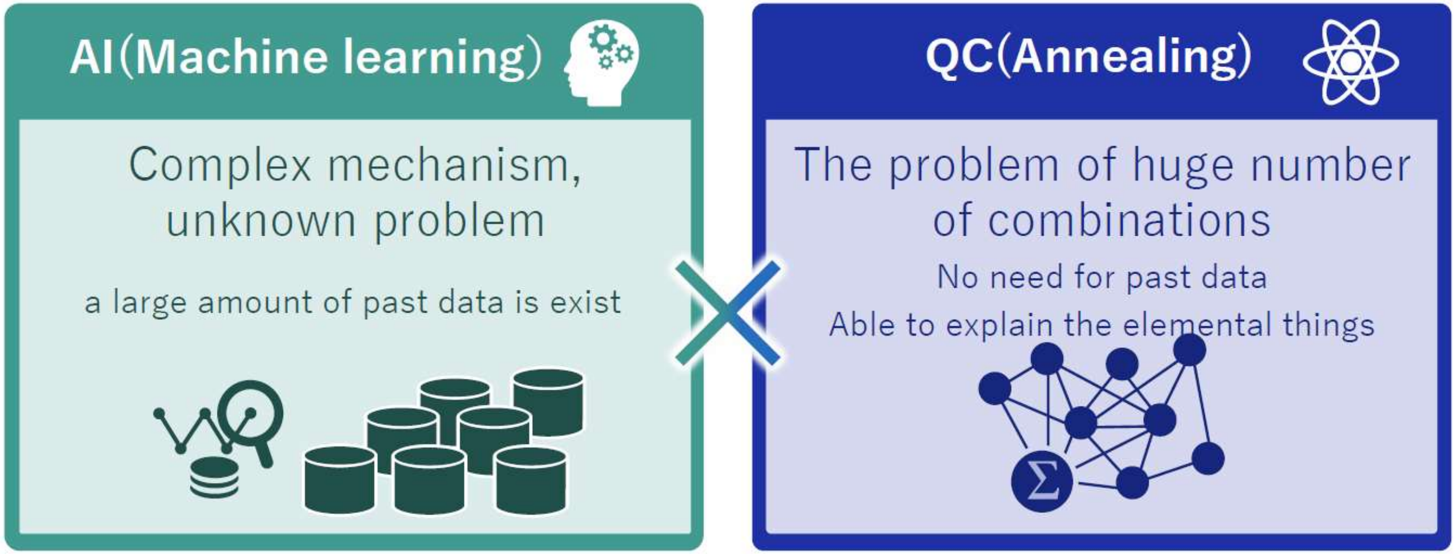


Not a single architecture optimal for all problems
Need for multi-architecture systems

Conclusions

- Applications with quantum annealing / computing just started
  - A lot of potentials
  - Hybrid algorithms QA / HPC or even QA / HPC / AI are to explore

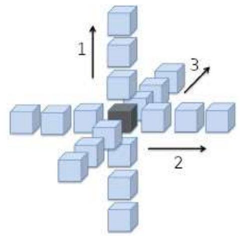
Both AI and QA are enablers for solving problems that are difficult with conventional methods



Conclusions

- NEC SX-Aurora Vector Engine is ready to address upcoming challenges
  - Easy and flexible porting (Native / Accelerator / Scalar modes)
  - Fruitful co-design projects with customers allow to identify critical enhancements

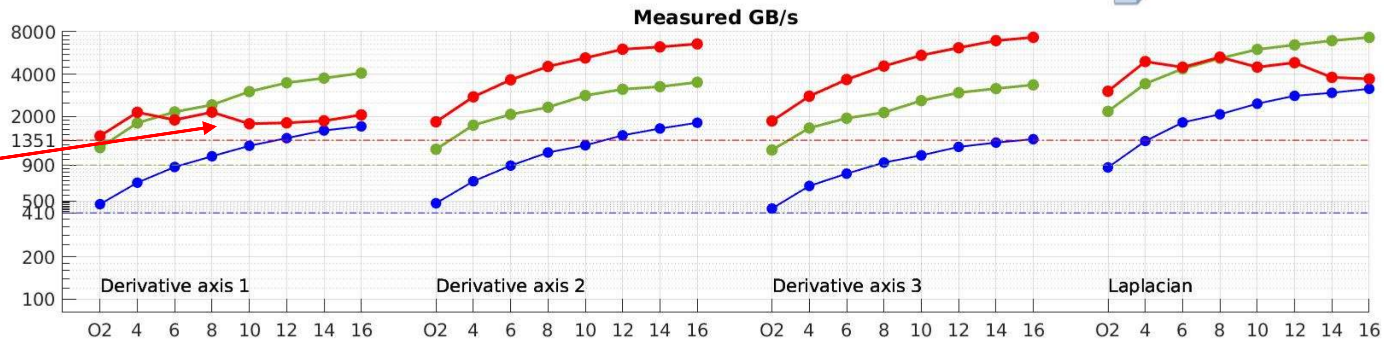
Example of co-design outcome for seismic imaging



Finite difference operators

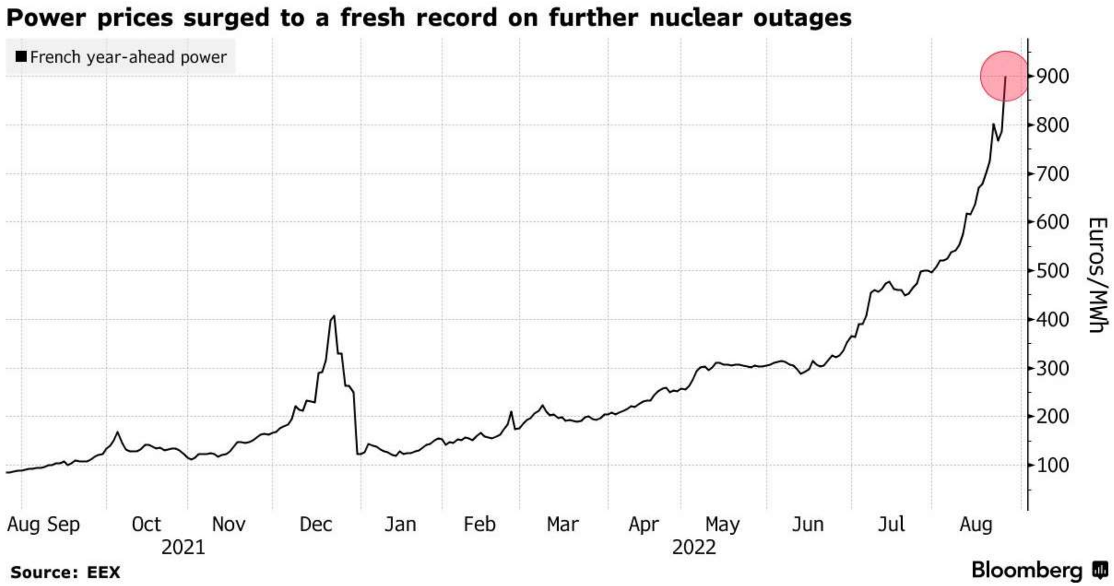
Optimization scheduled for Aurora Generation 3

Introduction to new instruction to increase performance on axis 1



Conclusions

- NEC SX-Aurora Vector Engine is ready to address the most critical challenge for HPC: energy
  - **High ratio computing efficiency / energy consumption**



Price of electricity in France (bloomerg.com)
10X in one year

June 2022 (top500)
For very first time, **EFlop/s is reached** (10^{18})
Frontier (DOE, ONL US)
1.1 EFlop/s with 21.1 MW

With current price in France

- 1 day → 456 000 € (455 000 USD)
- 1 year → 166 M€ (165.7 M USD)

Thank you very much for your attention

For questions or to get in touch contact me at **vincent.etienne@emea.nec.com**

\Orchestrating a brighter world

NEC creates the social values of safety, security, fairness and efficiency to promote a more sustainable world where everyone has the chance to reach their full potential.

\Orchestrating a brighter world

NEC